2011.07

PROGRAMMER 程序员

寻找技术领袖 P26



56 移动开发者俱乐部专辑

优秀移动应用团队 20 家

热点应用团队成功经验、创始人心得、产品运营分享。

22 图片报道

探访硅谷创新企业：软件工程师的乐园

52 WWDC2011 苹果大会感想

82 facebook 技术型领导

95 世界知名数据库专家 Raghu Ramakrishnan 访谈：谨慎使用 NoSQL

98 聚焦云计算

中国工程院院士李德毅：云计算实践分析

124 改变 Web 开发格局的新技术 node.js

ISSN 1672-3252

9 771672 325111 07

邮发代号：2-665　定价：15元

www.programmer.com.cn　www.csdn.net

Contents



P22

资讯

7 外刊速递

10 简讯

14 新产品新工具

16 程序天下事

热点报道

22 探访硅谷创新企业：软件工程师的乐园

24 硅谷直通车——离岸软件产品研发


P26

封面报道：寻找技术领袖

众多IT企业的技术负责人，决策着公司的技术选型和架构、技术团队的组织与管理，并凝成一股合力，决定了技术的时代潮流。这其中，总有一些耀眼的、具有领袖气质的技术领导者。本期封面报道，我们通过对八位技术领导者的访谈，以期找到塑造这个时代潮流的技术领袖气质，并且记录领袖们的成长、成功轨迹。

29 SAP中国研究院总裁芮祥麟：从千里马到伯乐

34 英特尔中国研究院院长方之熙：我与研究的不解之缘

38 微软大中华区CTO 张湘辉：技术成长的成功之路

41 神州数码信息服务集团副总裁谢耘：敢于挑战所有未知领域

44 Autodesk中国研究院核心产品及运营资深总监云浪生：成功是一种习惯

46 搜狗CEO王小川：不做规划 也要步步为营

48 豆瓣网首席架构师洪强宁：寻找技术味儿相投的人

50 英孚教育全球CTO Enio Ohmaye：重度技术体验者

移动开发者俱乐部专辑

52 WWDC感想

6月8日，2011年苹果全球开发者大会在旧金山召开，受到全球IT从业者的广泛关注。国内众多人士亲赴大会现场，给我们带来了WWDC 2011的精彩解读。

56 值得推荐的20家移动应用团队

数据显示，目前真正实现盈利的移动创业团队仅占25%左右。应用的立项、设计和研发、运营和推广这些环节均是导致一款应用能否成功的瓶颈，为此本刊特别在国内的诸多创业团队中，筛选出了“值得推荐的20家移动应用团队”，他们均在不同层面取得了成功，并在上述环节中都有可借鉴之处。

57 顽石互动创始人吴刚：走高品质的发展道路



技术领袖的模式

本期我们关注的对象是技术领袖。我们所说的技术领袖，指的是企业中负责把握公司及其产品技术方向的人。因此，技术色彩浓厚的创始人/CEO、CTO、总工程师、工程副总裁、技术总监、研究院院长、首席科学家等都可以纳入技术领袖的范畴。当然，最典型的还是CTO。

现在大家都耳熟能详的CTO一词，其实是在20世纪80年代随着研发型软件企业的兴起而出现的，又随着互联网大潮而普及。但是CTO一直没有非常明确的定义，对CTO的相关研究也很少。本期名人堂人物、曾经担任微软CTO的Nathan Myhrvold在一次访谈中表示，当年盖茨和他讨论新设的这个职位时，想来想去，连像样的先例都找不到，许多在技术领导方面做得很好的人都不叫这个头衔。他在微软的角色最后确定是关注技术发展，为未来做好准备。他这段职业生涯的主要成就是创办了现在声名显赫的微软研究院。

业内公认的最优秀的CTO之一Amazon的Werner Vogels，还专门为CTO的角色定义写过一篇文章。他引用Tom Berray的研究，将CTO分为四种模式：

基础设施的管理者 在大多数传统行业企业里，往往同时设有CIO和CTO。CTO将主要负责IT基础设施和运营：数据中心、网络、应用开发与维护、安全等等。CIO则负责如何在公司里应用技术。在这种模式里，CTO向CIO汇报。

技术先知和运营经理 在互联网和其他技术导向的企业中，IT是业务战略的核心之一。CTO需要负责决定如何通过技术实现业务战略。然后，他还需要负责实际实现技术。

面向外部的技术专家 在许多技术产品/服务厂商中，CTO主要作为客户与内部研发团队之间的媒介，负责很多客户沟通和市场研究工作，是产品组合研发过程中的主要影响因素。

思想家 这种模式的CTO主要关注未来，评估怎样在公司内部使用技术开发新的业务模式和产品线，如何抢在竞争对手之前使用颠覆性技术改变市场格局。他的工作包括技术趋势、竞争分析、原型研发、规划等等。

6月IBM度过了百岁生日。在竞争残酷的IT业，IBM经历了大型机、小型机、PC、互联网几个时代，今年市值还超过了新生代的Google，的确是一个奇迹。其始终在技术和研发上投入，有多位技术领袖持续观察技术趋势并不断在公司战略上做出调整，肯定是最重要的原因之一。

我们正进入一个技术变革的周期，其影响史无前例，技术领袖的作用也愈加重要。2009年美国总统奥巴马还任命了有史以来第一位国家CTO——Aneesh Chopra。Chopra与国家CIO Vivek Kundra共同推动开放政府计划，推动政府IT基础设施全面拥抱云计算，某种程度上为美国近年来IT领域创新层出不穷奠定了基础。

技术领袖有多重要？在Facebook的发展历程中起过特殊作用的前工程总监黄易山在“技术型领导”一文（见本期）中总结，领导是否具备较强的技术背景，往往关乎一个公司的成败。他认为Facebook之所以能击败MySpace等竞争对手，Mark Zuckerberg本人的技术功底以及公司始终保持了强烈的工程师文化和技术上的卓越，是最重要的原因。

中国需要更多优秀的技术领袖，我们期待着新一代技术英雄大量涌现。



邮件：liujiang@csdn.net
新浪微博：@刘江CE
欢迎大家交流反馈，欢迎投稿、批评、建议和挑错

Contents



P56



60 木瓜移动：从平台到孵化器
——木瓜移动创始人沈思专访

62 手机网游是移动应用的第一个热潮
——掌上明珠创始人武春雷专访

64 移动应用优秀团队推荐

68 从数据看iOS移动应用开发
本文特邀同步网络创始人、91手机助手作者熊俊，结合iOS以及移动设备的最新分布数据、App Store 上的应用现状，向您解读iOS移动应用的开发和推广策略。

71 雅虎重拳出击移动数字传媒
——雅虎首席产品官布莱克·欧文专访

72 与平台共舞，Opera的成功之道
——专访Opera中国研发团队

74 Android市场将持续爆发
——点心OS软件架构师梁泉专访
Android 获得成功的关键因素是什么？目前有哪些软肋？正面临哪些竞争对手？来自点心OS的软件架构师梁泉为您作深度解读。

76 手机游戏：崇尚简单至上和零碎娱乐的永恒价值

80 新技术、新平台、新境界

工程管理

82 技术型领导
Facebook前工程总监黄易山撰写了系列文章，很好地总结了Facebook卓越研发文化中的宝贵经验。

84 解读光谱化开发模式



P126

TUP

86 详解互联网产品开发中的“快”字诀
本文从产品、运营、技术、管理四个方面，诠释了腾讯互联网产品研发中贯彻的价值观——“快”。

一分钟先生

90 创业者如何打动投资者
从投资者、已获投资的创业者、在路上的创业者这三个角度，分享如何打动投资者的感悟与心得。

2011年7月刊　总第225期

Contents

主管：中国社会科学院
主办：中国社会科学院文献信息中心
出版：《程序员》杂志社
网址：http://www.programmer.com.cn
国际刊号：ISSN 1672-3252
国内刊号：CN11-5038/G2
邮发代号：2-665
广告经营许可证号：京东工商广字0188号

总编：黄长著　Editor-in-chief：Huang Changzhu
社长/常务副总编：张悦校　President：Zhang Yuexiao
副社长：蒋 涛　Vice President：Jiang Tao
编委会：黄长著　张悦校　陈洋彬　蒋涛　曾登高　刘江
Editorial Member：Huang Changzhu　Zhang Yuexiao　Chen Yangbin
Jiang Tao　Zeng Denggao　Liu Jiang

执行主编：孟迎霞　Executive Editor-in-chief：Meng Yingxia
编辑部主任：常政　Director：Chang Zheng
责任编辑：董世晓　高松　郑柯
Editors：Dong Shixiao　Gao Song　Zheng Ke
特邀编辑：方梁　高昂　赵健平　吕娜　卢鹄翔
Contributing Editors：Fang Liang　Gao Aang　Zhao Jianping　Lv Na
Lu Dongxiang
美术设计：纪明超　Art Designer：Ji Mingchao
美术编辑：林象海　Art Editor：Lin Xianghai
流程编辑：李雅楠　Coordinator：Li Yanan
Tel：010-64351458
Email：editor@csdn.net

发行部　Distribution Dept.010-64351431
Email：sales@csdn.net

广告总代理：北京创新乐知广告有限公司
Sole Advertising Agency：Bejing CSDN Co.,Ltd
Tel：010-64376055
Email：ad@csdn.net
Marketing Dept：010-51661202（ext 149）
Email：market@csdn.net

读者服务部
Readers service Dept.
网上订购：http://dingyue.programmer.com.cn/
读者信箱：reader@csdn.net
地址：北京市朝阳区广顺北大街33号院1号楼福码大厦B座12层
Address: B-12th Floor Fairmont Tower NO.33 Guangshun North street,
Chaoyang District,Beijing
邮政编码：100102
电话：010-64351436
传真：010-64348545

法律顾问：北京中润律师事务所 王杰
Law Consultant：Beijng Hengsheng Lawyer Firm
印刷：北京盛通印刷股份有限公司
Print: Beijing Shengtong Printing Co., Ltd.
出版日期：每月1日
Publication Date：the first day per month
零售价：RMB 15.00元　新台币 390元　HK $ 35.00（港、澳）
US $ 9.00（海外）
Retail Price：RMB 15, NT$390, HK $ 35.00,US $ 9.00

本刊文章版权所有　未经许可不得转载发现装订错误或缺页，请将杂志寄回本刊读者服务部，即可得到调换。

聚焦云计算

95 谨慎使用NoSQL

——知名数据库专家Raghu Ramakrishnan访谈

Raghu Ramakrishnan是Yahoo! Labs搜索与云计算首席科学家、副总裁，世界知名数据库专家，ACM和IEEE Fellow，最近本刊记者对其进行访谈，就如何成为一名计算机科学家、技术者创业需要注意的问题以及云计算和数据库未来的发展方向等话题进行了广泛探讨。

98 云计算实践分析

李德毅院士对我国云计算实践做了三方面分析：云计算给软件园带来的变化、传统电信行业的纠结、云计算实践的试点。

102 HBase性能深度分析

HBase作为BigTable的一个开源实现，随着其应用的普及，用户对它的性能数据愈发关注。本文将为您揭开HBase性能测试的一角，邀您一起参与到对云计算模块性能调优的深度思考中。

105 云时代的企业应用数据挖掘

108 架构师接龙：139说客李祎 VS. 人云科技吴朱华

调试之剑

112 经典阅读器的经典死锁

作者调侃“经典的软件，发生的死锁也很经典”，详细描述了著名的PDF阅读器存在的一个死锁问题。

并行编程

116 常见多线程Bug分析

在编写多线程程序时，最困难的一个环节恐怕就是调试多线程Bug。本文介绍了常见的两类多线程Bug及其解决方案。

工具点评

120 十年一剑——Drupal 7

124 改变Web开发格局的新技术node.js

程序春秋

130 Mac OS X背后的故事（三）

——Mach之父Avie Tevanian

126 Geek

128 图书上架

134 名人堂

136 漫画与幽默

Wired

2011.06



Y Combinator，初创公司的新兵训练营

2005年3月，Paul Graham回到了自己的母校哈佛大学，为本科计算机俱乐部作演讲。当谈到创业公司的问题时，他突然心血来潮，确定了创办Y Combinator（以下简称YC）的想法，为有潜力的创业公司提供指导以及资助，并获取其股份作为回报。

YC为初创公司提供的不仅仅是起步平台，还包括宝贵的经验以及潜移默化的影响。在Graham的指导下，YC每年举办两次为期三个月的“新兵训练营”。每支获准加入的团队都能获得11000美元的种子基金，另外团队的每位成员额外有3000美元的资助。

作为交换，YC将拥有这家创业公司一小部分的股份，通常是6%～7%。这13周的时间里，成员们能够与行业里最有实力的投资者和创新者对话，从而让自己处于萌芽状态的想法变成成熟的企业。最后，在演示日上，他们将向世界上最顶尖的风险投资公司和天使投资人展示他们的计划。

过去五年，涌现的很多优秀创业公司中，不乏接受过YC训练营培训的成员。Dropbox、Reddit、Loopt以及Scribd的创始人，都是Graham发掘和培养的。Graham的慧眼独具，吸引了科技博客关注的目光，也为这些创业公司赢得了投资人的青睐。

在六月中旬结束的为期数天的“面试日”中，共有120组创始人带着梦想接受YC的考验。他们要在十分钟内说服Graham的团队，赢得2011年冬季训练班的准入资格。本次面试日，获准通过的创业公司有43家，共99人，较之前有所增加，2011年冬季训练班将是有史以来最庞大的。

但是，YC也引起了天使投资人的不满。在YC出现之前，天使投资人对创业公司有很大的影响力，创业公司要挖空心思地吸引投资人的注意，并且拿大部分的公司股份作为条件换取资助。而现在，投资人则需要争抢投资创业公司。

另外，虽然YC培养了一批成功的公司。但也有一些公司没能活下来。过去的六年中，大约有四分之一的公司夭折，还有很多经营惨淡。尽管如此，这样的失败率在硅谷的新公司中已经是相对较低的。

Twitter联合创始人：撼动信用卡行业

2007年，Jack Dorsey作为联合创始人创办了Twitter，使用户能够快捷地广播最新动态给自己的朋友。目前Twitter已经成为拥有2亿用户的沟通平台。他的新公司有着相似的目标：通过一个能够外接在iPad和智能手机上的小小的读取装置Square，让每个人都能接受信用卡支付。目前已经有几十万公司注册了这项服务，仅在2011年第一季度，就处理了6600万美元的交易额。这家创业公司同时也在建立一个可供客户挖掘利用的财务信息数据库。它能够记录每笔通过信用卡读取器完成的销售，为公司提供诸如一周中销售量最大的日子或者某项产品的平均价格之类的信息。这一类数据分析的强大力量，帮助Square在一月份的第二轮融资中获得了2750万美元的投资，公司的估值达到了两亿四千万美元。四月份，VISA也投入了一笔数额未公布的资金。

本期的《Wired》杂志，对Jack Dorsey进行了一次访谈。在被问到如何解决用户对Square的认可度以及对风险的疑虑时，他解释说每一个Square卖主都要通过身份验证。在使用该项服务时，会有相关的验证机制验证使用者的身份。另外，每张信用卡本身就包含一些保护措施。Square有一个十人小组专门对风险进行监控，还会通过社交网络数据验证某个企业是否正规。Square试图在每次传输中有效地加入更多信息，从而降低欺诈和支付不成功的风险。通过降低整个系统内的风险，可以降低信用卡公司和银行的交换率，从而降低商户的成本，进而降低销售价格，改善整个支付生态环境。他还说到，在网络上，公司可以通过类似Google Analystics这样的工具来分析用户行为，并利用这些数据。而对于占世界商业94%份额的离线商户来说，没有切实可行的途径来获得这些数据。现在，Square可以为商户提供这些数据，供他们提取有用的信息。

Steven Levy：苹果的智能提升策略

苹果公司的掌门人乔布斯一直坚信苹果产品的完美品质，对于外围的扩展附件嗤之以鼻。但值得一提的是，iPad 2

发布后，引起最多反响的不是设备的诸多显著改善，而是它的Smart Cover。这件用来遮盖iPad 2屏幕的柔软可折叠的附件，几乎成了乔布斯展示iPad广告的明星主角。

Smart Cover的设计与这款平板电脑完美契合，其中嵌入了磁铁，可以让它固位于指定的位置，打开它时可以唤醒iPad，合上时则让iPad休眠。Smart Cover可以折叠成一个支架，并且提供聚氨酯和皮革两种材料供选择。Smart Cover经过强有效的抗体浸泡，经常接触有助于避免霍乱、疱疹和普通的感冒。

苹果公司对小到像Smart Cover这样部分的认真精神，很好地说明了这家公司如何战胜那些不懂得重视每个细节的对手。

苹果的附件制造商一直处于不利状态：因为直到公开宣布的那一天前，苹果公司从来不透漏任何一款新产品的规格。这些制造商现在不仅要和其他制造商竞争，还要和苹果公司本身竞争。

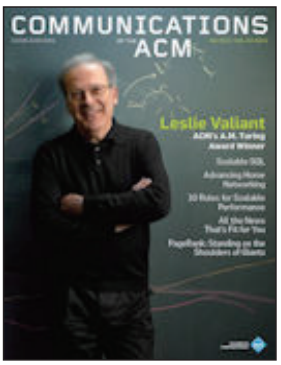
如果说iPad 2这件Smart Cover缺失哪项功能的话，那就是它没能起到保护整个iPad的作用。这点疏漏，让附件制造商们还能有一点喘息空间。目前，附件供应商着眼于“盖”与“套”的区别，并努力在第二类产品上做到最好。

也许乔布斯这次没有坚持，不情愿地默许自己心爱的设备存在瑕疵的原因在于，从长远来看，很难判断用户会选择iPad还是其他平板电脑。而加入智能盖之后，iPad 2的成功就更加万无一失。

借iPad 2的发布，乔布斯为自己的对手补上了不少枪。iPad目前占据了大部分的市场份额，而新版的发布会进一步加大这种优势。

Communications of ACM

2011.06



一生的学习者——Leslie Valiant访谈

哈佛大学工程与应用科学学院的计算机科学和应用数学教授Leslie Valiant，是2010年ACM图灵奖的获得者。他最著名的成就是提出了机器学习的“可能近似正确”（Probably Approximately Correct，PAC）模型，它为学习过程的研究提供了一个基本框架，并且导致了从人工智能到计算机视觉等一系列领域的重大进展。

Valiant在1984年发表的论文“可学习性的理论”中，开创性地提出了机器学习数学框架的基本思想，也就是我们熟知的PAC学习。

在访谈中，Valiant提到了自己创造PAC模型的初衷是为了定量化学习算法的有效性。他还提到了应用PAC提出的另外一种数学模型——弱学习（weak learning）。这项技术可以用来改善基于真实数据集的几乎任何一种基础学习算法的性能。

20世纪80年代，Valiant在分布式计算和并行计算领域做出了很多重要贡献。他建立了整体同步并行模型（Bulk Synchronous Parallel Model），为并行系统的设计创造了解析框架。Valiant认为并行机的根本问题在于很难用一个统一的模型一以概之。时序机有冯·诺伊曼模型作为软件和硬件之间的“桥接模型”。他致力于寻找一种适用于并行机的简单有效的桥接模型。目前，他提出了一种桥接模型——MultiBSP模型，以尽可能简单地捕捉并行机中的物理局限性，探寻并行计算的基本原理。

Valiant还致力于计算神经学的研究，在他的专著《思维的电路》（*Circuits of the Mind*）一书中提出了“神经接口模型”（neuroidal model），用来描述神经元层次上基本计算任务是如何进行的。他提到，在神经元层次上，大脑和计算机有着明显的不同，但是在更高的层次上，我们可以找到一些可以用计算术语来描述的基本要素。大脑存在一些通用计算特征，我们有望在不久的将来破解它们。

在访谈中，Valiant还提到了“鲁棒逻辑”系统，用来研究包括人脑在内的计算设备是如何执行任务的。

印度的计算机行业

现代计算机科学和印度有着很久的渊源。20世纪50年代，印度开始了计算机研发，孟买的塔塔基础研究所和加尔各答的印度统计研究所自主制造了计算机，主要应用在科研领域。60年代起，商业计算中心开始外包业务，进行电话和电力计价、航空规划等等。但是计算机科学迅猛发展的70年代，印度因为在学术和工业领域的科研投入不足，严重落后。

20世纪80年代，计算机化风潮快速席卷全球。大大小小的机构组织都开始使用计算机来提高效率。随着软件开发的兴盛，催生了对程序员的大量需求。

印度有大批受过软件训练的年轻人，并且具备这几个必要条件：青年、英语训练、对学习的渴求和就业需求。所以印度毫无疑问地要在这个阶段扮演重要的角色。千年虫问题带来了史无前例的需求，印度软件行业顺势而上，迅速发展。20世纪90年代，该行业对印度GDP的贡献几乎为零，今时今日已经占到了GDP的将近7%，创造了超过三百万的直接就业岗位。

尽管软件行业迎来了繁荣时期，但是印度明显忽视了在高质量学术研究上的投入。印度有三百万IT专业人员，一百万计算机专业本科生，但是每年培养出的高质量博士只有不到一百人。而且虽然计算机科学研究者中有很多印度人，但是大多都不在印度本国。

目前，印度的十二亿人口中，处于1～15岁年龄段的有四亿，处于15～35岁年龄段的也有四亿。所以，印度有条件也有义务为计算机和相关领域培养未来的知识劳动力资源。

印度面临的挑战包括提高计算机本科教育的质量，以及吸引更多人进行计算机研究。近十年来，虽然大学入学人数有所增加，但教育质量没有跟上。另外，增大高水平研究上的资源投入也势在必行。

选择“易操作”数据存储库可扩展性能的十条原则

自1970年，数据的关系模型诞生以来，就成为了数据库管理系统的最好解决方案。过去五年里，针对Web 2.0和其他应用，涌现了很多可扩展的SQL和NoSQL数据存储库，用来扩展单服务器关系数据库管理系统的容量。

本文的重点主要在易操作（simpleoperation）数据存储库上。目前只有很少新的、非GPTRS系统针对这个市场提供了可扩展性。可以将它们不严格地分为四类：键值对存储、文档存储、可扩展记录存储和SQL数据库。其中，前三类系统很大程度上推动了NoSQL运动。

这些系统的诞生，背后有很多推动因素。其中部分是因为对关系模型的不满，或者说关系数据库管理系统的“笨重”。另一部分是因为大型网络性能的需要。这些新功能解决了目前存在的一些问题，迎合了新需求，问题是如何从这么多新系统中选择最适合客户应用要求的那个。

这些数据存储库的区别主要在于一致性保证、每服务器性能、读和写负载的可扩展性、服务器失效的自动恢复、编程方便性以及管理便捷性。

文中列举的十条原则，详细描述了一个易操作数据存储库应该具备的特性。

一、具备“非共享”可扩展性。数据库管理系统可以工作在三种硬件架构上：多进程共享（SMP），运行在磁盘簇上的数据库管理系统，以及各节点之间不共享内存和磁盘的非共享方式。

二、可以使用高等语言，但不可因此有损于性能。高等语言可以减少代码量，提高代码可读性，更易于理解，可以减少代码维护工作量。

三、谨慎地使用内存数据库。如果一个数据库达到数T字节的规模，用户必须考虑内存部署问题。

四、对易操作可扩展性来说，高可用性和自动恢复是必须具备的。

五、一切都在线实现。从用户的观点来看，系统应该永远都有效，永远都不下线。

六、避免多节点操作。最大限度上地避免多节点操作，包括必须分成多块的查询和要满足ACID性能的更新。

七、不要试图自行建立ACID一致性。 如果需要ACID，那么就选择提供ACID的数据库管理系统，不要自己编码实现，以避免碰到分布式事务的天花板。

八、管理便捷性。一定要亲自进行概念验证（proof-of-concept）以便考察其即开即用性。

九、关注节点性能。虽然说线性可扩展性很重要，但是节点性能也绝对不能忽略。节点的性能越快，需要的节点数就可以减少。

十、开源让你更好地掌控未来。避免“供应商弊端”的最好途径就是使用开源产品。



IBM董事长彭明盛在IBM百年庆典上发表演讲

IBM百年

6月16日，IBM满一百岁了。从打孔卡到S/360大型机，从制表机到浮点运算，从DRAM到超级计算机，从FORTRAN到SQL，蓝色巨人走过了一段独一无二的历史。

近些年来，IBM发力软件服务领域，再度崛起。这家科技巨头在一百年的风雨历程中形成了自己的文化：重视培养内部领导力，投资技术研发，深思远虑。

也许一个企业能否存续一百年真的不那么重要（对变动不居的互联网公司尤其如此），但是能将自己的企业文化一以贯之，这才是我们纪念IBM百岁的意义所在。

淘宝拆分为三

在马云就支付宝问题，多次表示自己在“做一个艰难的决定”的同时，阿里巴巴旗下另一品牌淘宝被一分为三：一个类似于eBay的消费品网上市场（淘宝网），一个大型零售商和国际知名品牌的网上商城（淘宝商城），以及一个电商购物搜索引擎（一淘网）。对此马云说，“为了更好适应今天行业的快速发展，集团决定提升‘大淘宝’战略为‘大阿里’战略”。这一举措无疑激起了投资者对淘宝近期将会上市的期望值。

包括但不限于以上动作，可以看出阿里巴巴也正试图从CEO、COO受内部欺诈案调查而引咎辞职的事件中恢复。


马云又一次冲上舆论的风口浪尖

Bitcoin来了？

Bitcoin的最初概念由Satoshi Nakamoto提出，它不依赖于特定的中央发行机构，而是使用遍布整个P2P网络节点的分布式数据库来记录货币的交易，使用密码学的设计来确保货币流通各个环节安全性。这些设计确保Bitcoin的价值不可能被人为操控，同时避免了通货膨胀。但有大批人士对Bitcoin持怀疑态度，而Bitcoin最大的交易市场Mt. Gox在6月20日发生一连串近50万Bitcoin极低价格的巨量交易，造成市场崩溃的事件，也表明Bitcoin要想达成最初的设想，还有相当长的路要走。

传统纸质书籍或将消失

KK撰文谈及传统纸制书籍正在消亡，电子书将在本世纪内成为廉价丰富的大众消费品。同时，Internet Archive的创始人Brewster Kahle决定保存每一本Google和Amazon已经扫描成电子书的纸版，数目预计将达到1000万册书，其意义在于：数字版存在忠实原著和准确性的问题，如果有图书出现错误，只有原始副本才能提供可靠参考。但世界上从来不缺乏唱反调的人，自由软件基金会主席RMS发表了文章“电子书的危险”，呼吁消费者拒绝购买电子书，除非它们“尊重我们的自由”。

“上网本已接近消亡。”

——联想总裁兼COO Rory Read表示，在未来几个月内，联想将在全球范围内推出十英寸Android平板电脑，以取代大势已去的上网本。

“苹果最有可能生存一百年。”

——经济学人撰文称，继IBM之后，Apple、Amazon和Facebook有可能生存一百年，甚至更久。而其中Apple以简单、成功的理念，“利用最新的技术，进行简洁出色的包装，然后高价销售”，成为最有潜力的候选。

“甲骨文的沉默可谓此时无声胜有声，我们对此非常失望。”

——HP因支持Intel Itanium芯片的问题将Oracle告上公堂，此前HP曾向Oracle发律师函，要求后者改变停止为Itanium芯片开发软件的决定。HP首席发言人Bill Wohl在谈到Oralce对律师函的反应时，作如上表示。

“这个市场不缺商品、不缺卖家，移动电子商务最多在三年内就可以赶超传统电子商务十年的积累。”

——淘宝网副总裁邱昌恒预言移动电子商务将迎来爆发性增长。

“要想改变世界，不能依靠一时的灵感或智慧，而是需要多年的练习和努力。一切真正了不起的事物都需要许多努力。”

——Facebook CEO Mark Zuckerberg在美国加利福尼亚州Menlo Park的一所中学发表演讲，指出成功没有捷径。

“我的所作所为并不想对任何人造成伤害，我只是想通过这种方式引起苹果的注意，我热爱这家公司。或许今后我能够有机会同苹果CEO Steve Jobs会面。”

——苹果“粉丝”和业余开发者Mert Erdir声称，他通过破解iOS 5的Voice Over功能而成功安装该操作系统，并为自己的行为初衷辩解。

“Google一直自夸的具备延展性的软件架构已经过时了。”

——Google前工程师Dhanji R. Prasanna在博文中，解释自己为何要离开Google，他认为Google的软件架构已经过时，尽管Google的硬件架构依然是完美的艺术品，但为搜索引擎和爬虫设计的软件至少落后了10年。

CSDN最受关注文章

2011年4月26日~5月26日

01 浙江大学获ACM-ICPC 2011冠军

ugmbbc

被誉为“计算机领域奥林匹克竞赛”的ACM国际大学生程序设计大赛，是世界上规模最大、历史最长、最具影响力的大学生国际级计算机类赛事。在今年举行的第35届ACM国际大学生程序设计竞赛全球总决赛中，浙江大学荣获全球总冠军。

02 2011年美国IT从业人员的薪资调查

佚名

据2011年ComputerWorld的行业薪水调查显示，今年一些IT从业人员比其他同行拿到了明显丰厚的薪水。报酬优厚前五名的行业为：商业服务、旅游管理、交通运输、人工制造和防御工事/航空航天。奖金最高前三名的行业为：交通运输、人工制造和零售业。

03 比尔·盖茨老照片：13岁走上编程之路

宗和

介绍了盖茨这位技术天才（13岁开始编程、20岁不到写出BASIC语言）、商业天才（31岁便成为世界首富、并蝉联十二载）的生平。特别有意思的是近期发表的盖茨的老照片，展现了他与朋友合作、毕业典礼，甚至被捕入狱，平凡而不为人知的一面。

04 用移动互联网精神做MeeGo

付江

Intel在MeeGo上花的气力有目共睹。尽管开放源码，且不受任何单一企业支配，还可以跨平台和跨设备移植，但令人遗憾的是，MeeGo始终没有利用好这些竞争优势。终端匮乏是MeeGo生态系统短板。也许只有加速落实在终端设备的模式策略，MeeGo才更值得期待。

05 C语言之父Dennis Ritchie访谈

陈秋歌

Dennis Ritchie谈及如果重新设计，C语言需要改进的地方，并为自己的原有设计作出解释：内在一致性的要求。还提到了自己最喜欢的Unix实际应用，比如Kindle和Android等智能手机。最重要的是，他认为当年AT&T因为思维定势而对早期互联网太多的机会视而不见。

06 百度副总裁史有才离职

乐天

史有才是百度初创团队成员之一，参与百度渠道建设达十年之久。他在离职邮件中表示，百度十年，自己不但获得了财务自由，还拥有更广阔的视野和简单可依赖的价值观，并回顾了当年加盟百度时的情景，以及十年间搜索推广市场的变革。



1977年，比尔·盖茨因超速驾驶被捕

07 那些被微软“干掉”的著名公司

阳淼

微软的进攻性极强，这个IT巨头依靠观察、模仿、消灭等竞争手段获取商战中的制高点，被它伤害过的企业可以列出长长的名单（不乏Lotus、Novell、Borland等知名公司）。这些名单如今看来已经过时，但对无数正和对手角力的公司，这都将是宝贵的经验。

08 IBM历经十五载市值终超微软

马乔

这是自1996年4月以来IBM的市值首次超越微软。2000年爆发的互联网泡沫是这场长期竞争的转捩点，因缺乏新的增长点，微软股价停滞，而IBM则重新为自身设计了发展战略，十年后，IBM重整百年荣耀，但它们前面都还有一个更强劲的对手：苹果。

09 苹果失误造就Android的成功

张楠

诺基亚CEO Stephen Elop称正致力成为应用开发领域第三家主要厂商，成为Windows Phone手机的首家OEM厂商，与运营商结为合作伙伴，并谈及是苹果失误造成Android的成功，苹果的方式让很多人难以接受，Google才得以介入并创造了Android。看得出这也是为自家打气。

10 高效的面试方式：结对编程

Rod Hilton

面试是技术研发企业永恒的话题。什么样的面试方式才是最高效的？资深Java程序员Rod Hilton在博客文章中介绍了他们公司的做法：结对编程。此文在Reddit上引起了强烈反响。有人提醒其中的法律问题，还有人认为这种面试会耗费更多资源，未必值得效仿。

全球IT专利排行榜

®品牌 2010年获得的专利数量

科技公司一直是推动创新进步的主要动力。不过，尽管IBM和通用电气占据专利数量的第一位和第三位，但佳能、日立和三菱等日本企业却占据了15大创新企业前十位的大多数。具有突破意义的专利不仅可以推动世界进步，也可以为这些公司带来数十亿美元的收益。

01 ®IBM 5866项

IBM最著名的发明包括磁条技术（例如信用卡所用的）、计算秤、通用产品代码（或者叫条形码），以及自动计分。

02 ®Canon 2551项

佳能最著名的发明包括率先在美国上市的IVSB相机、10键电子计算器、首款自动曝光的单反相机和个人复印机等。

03 ®通用电气 1222项

通用电气的发明包括碳丝灯泡、X射线机、电扇、首款密封冰箱、便携式空调，以及数字闹钟收音机。

04 ®日立 1447项

日立的发明创造包括5马力发动机、首个大型直流电动机车、最小的无接点芯片，以及可以用作导游的互动式机器人。

05 ®东芝 2212项

东芝最著名的发明包括日本首款电动洗衣机和真空吸尘器、首款彩色视频电话，以及首款个人笔记本电脑。

06 ®三星 4518项

三星的发明包括首款微波炉、首款8毫米录像机，首款1G闪存原型，以及首款可以用作MP3播放器的手机。

07 ®索尼 2130项

索尼的发明创造包括Walkman、首款便携式CD播放器、首款非投射便携式晶体管电视，以及日本晶体管收音机和磁带。

08 ®松下 13项

松下最著名的发明包括超级电熨斗、电子搅拌机、磁带录音机、电冰箱等。

09 ®NEC 652项

NEC的发明包括普通电池板、手机音乐播放器的芯片、首款数字信号处理器，以及微型电脑。

10 ®富士通 1276项

富士通的发明包括日本首个海底同轴电缆系统、同款42英寸彩色等离子显示屏、笔记本用的可降解塑料，以及1968年开发的一款具备微处理器系统的电脑FACOM 230-60。

IBM Innovate 2011

2011年6月3～7日，IBM的（Rational）软件技术盛宴Innovate 2011在美国奥兰多市成功举办。IBM高级副总裁兼软件与系统集团总经理Steve Mills、IBM软件Rational全球总经理Kristof Kloeckner等高管登台演讲，为全球技术人员诠释了软件生命周期管理、开发者职能和角色的演变、未来软件开发模式以及IBM的最新策略。今年大会的主题围绕软件能给企业带来的核心竞争优势展开，关键词是协作、整合、敏捷、软件计量经济学。

第一天的演讲中，Kristof表示，Rational软件发展这么多年，有几个关注点是始终不变的，即协作、整合、优化、高效。他特别强调了协作开发的内涵：如今软件工程中的所有步骤都需要团队共同参与，不光是开发者，更重要的是实践者(应用者)，以及设计人员、测试工程师、架构师等都应该一起完成设计、开发、部署和使用软件的过程，而不再像以前那样采用割裂式、流水线般的研发流程。



关于敏捷开发，他认为核心就是重复使用，他以Rational软件为例，即使在跨众多不同行业的大型企业应用中，也有很多基础模块是重复的。这也是Rational提供给用户的中心议题——模块化、反复的重新使用。

大会第二天， Steve Mills登场，他回顾了IBM软件集团的成长历程，再次强调了IBM的软件策略：不做直接面向消费者的应用软件，而聚焦在企业级软件市场（尤其是大规模用户的平台级软件）。

他通过数据分享到，现阶段绝大部分IT投资都用在管理已有的IT设备上，靠优化系统和软件来提升效率和性能迫在眉睫。Steve Mills给了三个建议：硬件资源整合、消除软件（功能）和数据的冗余部分、改进交付方式。

正值IBM诞生一百周年之际，在IBM Innovate 2011上，无论是IBM高管还是技术粉丝，都表现出了对创新发展的巨大热情，可以看出蓝色巨人仍然活力充沛。

HTML5 in China

2011年6月10日下午，“HTML5 in China暨首届HTML5原创游戏大赛启动大会”在北京京仪大酒店举行。此次活动由HTML5研究小组与创新工场、W3C中国联合主办，CSDN、游戏邦协办。

会议由创新工厂创始人李开复做开场演讲。李开复强调，“互联网的未来是属于草根的”。而推行互联网标准，将铲平由大公司和商业巨子把持的世界，使之变得更加平坦。

李开复随后从三个方面论述HTML5发展的必然：HTML5 不是简单格式的升级，其使命是打造帮助Web发展更成熟的平台；HTML5以标准化的方式在设备多元化时代，提供了更好的服务；打破互联网孤单现象的统一平台的基础，将仍是HTML5。

随后，W3C HTML5 WG Lead MikeSmith、华尔街HTML5资深专家殷唯一，以及四大浏览器厂商（微软、谷歌、Opera、火狐）开发人员分别就HMTL5的诸多特性和Web技术的发展做了精彩演讲



在全场观众、嘉宾和各地HTML5小组负责人的见证下，

整场大会的重点，HTML5原创游戏比赛启动仪式拉开帷幕。

据了解，此次大赛作品征集时间为2011年6月10日～9月30日，广大HTML5游戏开发者，可将原创游戏发至大赛官网，参与比赛。每一位参加比赛的开发者，在展现创作才华、增加互动交流的同时，还能收获发展自身，推广作品的机会。

大会的后半段，分设了HTML5游戏和非游戏两个会场。

游戏专场里各路专家介绍了HTML5游戏创意、HTML5联机游戏、HTML5云端游戏、HTML5新平台等扎实有料的实战经验。

非游戏专场上，则有演讲嘉宾做了HTML5应用案例的演示和互动分享，介绍了基于微博、IM、云和电子商务的HTML5技术等内容。

随着HTML5原创游戏比赛启动，将有越来越多的人参与到应用HTML5的开发中，而HTML5还在不断发展中，随时都有更棒的技术涌现，这一股强大的合力，必将推动Web平台朝着更宜人的方向发展。

微软Tech·Ed 2011参会感悟

5月16~19日，五名从7宝7谋活动中胜出的中国Windows 7谋士参加了在美国亚特兰大举行的Tech·Ed 2011大会。Tech·Ed即微软技术大会，是微软集合世界各地的开发人员和IT专业人士，提供技术分享，社区互动和产品的评估资源的最大规模的技术盛会。

本次大会以“云计算，设备和开发者”为主题，集中讨论了微软的云产品。在主题演讲中，微软服务器和工具营销副总裁Robert Wahbe重点介绍了云计算和虚拟化的机会，并通过视频展示了美国杨百翰大学的研究团队开发的基于USB设备的便携式超声系统，它可以连接到任何一台笔记本电脑中，将图片上传到云中。此外，微软还表示正在开发Visual Studio VNext，重点是缩短新应用从概念设计到部署的开发周期时间。在本次Tech·Ed上，微软还携硬件厂商提供了涵盖游戏、商业智能、统一通信、服务器、数据库等尖端技术展示。

在微软Windows 7宝7谋活动中获选并赴美国参会的五位谋士，向本刊读者分享了他们的参会感悟。

神州数码（中国）有限公司微软事业部工程师、讲师苏华：通过本次大会，我了解到Microsoft“云+端”的战略布局已基本成型。现在全世界范围内Windows Azure Platform总共构建了六个数据中心，即美国两个、欧洲两个、亚洲两个。

广州某传媒公司主任编辑李韬：很明显，Tech·Ed 2011大会表明微软仍在大力部署云计算，仍期望建立一套以云计算为核心的庞大的应用软件支撑系统，以便在下一个时代中的霸主地位不被轻易撼动。一位来自中国的技术工程师称，此次大会并没有展现出明确的新技术方向，当前微软处于产品“青黄不接”的时期，一切似乎都需要明年面世的Windows 8来揭开谜底。



成都大学软件工程专业在校生李贵江：大会主题为Cloud，会上详细讲解并演示了众多微软云相关产品和虚拟化产品System Center、Windows Azure、Hyper-V、SharePoint及数据库新一代产品SQL Server 2008 R2。其中Windows Server 2008 R2、Hyper-V及虚拟化部署是我最关注的云计算产品。

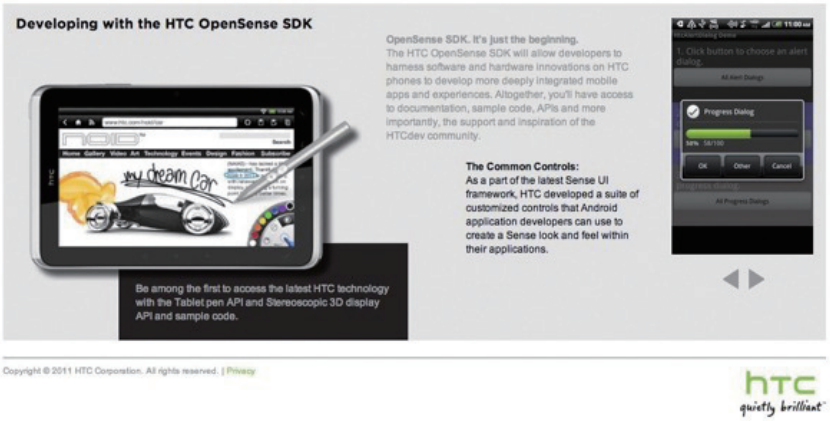
济南历下区教育局IT工程师杨帆：除了云计算，会议还展示了Windows Phone 7和Kinect等新技术。云计算仍将是微软未来几年的战略产品；Windows Phone 7带来了全新的用户界面和手机操作体验；Kinect的出现彻底颠覆了传统的游戏控制方式，无需任何手柄只靠身体动作就能实现游戏操控，非常酷。

河南华坤科技有限公司副总经理苏繁：本次Tech·Ed几乎富含了微软所有产品和技术，我对其中的Windows MultiPoint Server 2011最感兴趣，并有幸体验了这一产品，它只需一台多显卡的PC和一个USB hub即可实现一机多台使用，但有一点，如果所使用的USB hub没提供音源接口的话，声音将无法传递到每个终端上。

HTC公布第三方开发者OpenSense平台

HTC公布了其面向第三方开发者的OpenSense平台，目的是为他们的HTC Sense界面开发相匹配的应用。 HTC Sense是HTC开发的用于Android的自定义界面，它重新设计了主屏、锁屏界面、Web浏览器等，并添加了大量的首页小工具。

公司还启动了一个新的开发者门户网站HTC dev用于介绍他们的最新计划，不过暂时还没有什么开发工具出现在网站上，预计在数周内就可以出现。

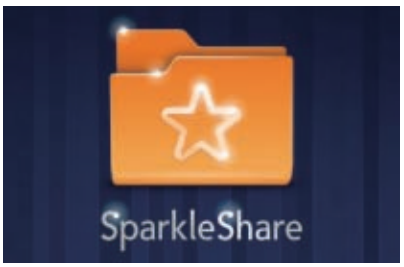


JavaScript MP3播放器和PDF浏览器

HTML5和JavaScript正日益展现出取代原来需要插件的富应用的势头。Mozilla的一个团队正在开发基于JavaScript的PDF阅读器pdf.js，直接在标准的HTML页面上载入和渲染PDF文件，无需专门的阅读器。未来的Firefox预计将整合该应用。柏林的Music Hackday则开发了无需Flash的JavaScript MP3播放器jsmad.org。

微软发布 Kinect SDK Beta 开发工具包

微软发布了运行在 Windows 7 环境的Kinect SDK Beta。该版本主要功能包括：原始传感器流，从深度传感器中访问原数据流，彩色摄像机传感器，四透镜麦克风阵列可以为开发者构建由Kinect传感器生成的低级别流。Skeletal跟踪器：可以用来追踪在Kinect领域视野中来回移动的一两个人的骷髅图片，这样会更容易创建手势驱动的应用程序。高级音频性能：具有较强的音频处理能力，包括抑制吵杂的噪声，消除回声；提供实例代码和文档，SDK 提供了100多页的技术文档，除了帮助文档，还包括SDK大部分实例中的详细演练。



云存储工具SparkleShare发布

SparkleShare是一款开源的云存储及文件分享工具，与DropBox等工具类似。该应用使用GitHub、Gitorious或个人GIT服务器作为存储空间。SparkleShare目前支持Linux和Mac平台。未来会支持Android 和 iPhone手机。

基于 HTML 5 的实时音频和视频通讯平台 WebRTC

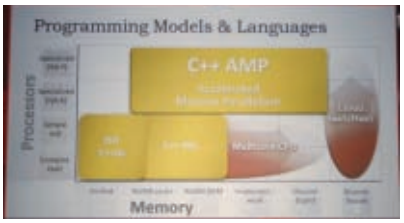
Google宣布推出开发者预览版的WebRTC，这是一个可让网友之间通过音频和视频实时交流的开放工程，只要是支持Real-Time Communications（RTC）的浏览器都可实现实时音频和视频聊天。

这个新的技术使用了HTML 5和简单的JavaScript API，开发者可以很轻松创建RTC应用，Google希望用这个开源工具击败微软的Skype和苹果的FaceTime。

WebRTC应该是基于Google在去年收购的GIPS技术，其实在收购之前GIPS就制作了Android、Windows Mobile和iOS版的VideoEngine，可实现跨设备的实时聊天标准。

Miro 4发布

开源跨平台视频播放器Miro发布了v4.0版，新的特性使它完全可能成为iTunes和Windows Media Player的替代者。开发Miro的Participatory Culture基金会执行董事Nicholas Reville表示，他们将证明开放媒体平台完全可以挑战封闭平台，如iTunes等。



微软宣布异构计算语言C++AMP

在AMD Fusion开发者峰会上，微软推出了C++ AMP （accelerated massive parallelism，加速大规模并行）。C++ AMP是微软Visual Studio和C++编程语言的新扩展，帮助开发人员充分适应高度并行和异构计算环境。它使用C++语言的语法，将随同下个版本的Visual Studio一同发布，微软宣布C++ AMP开放，允许其他编译器集成和支持。

Gaia Ajax 3.7发布

Gaia Ajax是一个强大的APS.NET Ajax数据库，它可以让你不用JavaScript编写ASP.NET网页前台。没有必要使用UpdatePanels或者是ScriptManagers，Webservices 也是可选的。

Amazon推出Mac Downloads Store

Amazon悄悄发布了针对Mac OS X系统的在线软件销售商店。在线商店现有超过250款软件，部分软件是Mac App Store所没有的，如Microsoft Office、QuickBooks、Photoshop Elements、Toast、Dragon Age Origins和The Sims 3。Amazon的OS X软件商店只是简单地提供标准程序的付费下载，特别之处是购买的任何一个软件Amazon 会在用户的Amazon Cloud中放上一个拷贝，用户可以随时根据需要重新下载。

Google +1按钮正式发布

Google正式推出 Google+1按钮，测试期间订阅邮件的用户均收到了Google发送的通知邮件。目前Google的搜索结果已经正式添加此按钮，当用户登录Google账户后进行搜索，均可利用条目右侧的按钮对搜索结果进行推荐。另外站长及博客主均可以安放Google+1按钮到其网页上，以方便用户进行推荐。



Webian Shell模仿Chrome OS

Mozilla Labs推出了一个新桌面概念原型Webian Shell，用浏览器替代传统桌面。目前该项目还仅是带Dock的全屏浏览器。它运行在现有的操作系统上，不能控制硬件和系统功能。目前支持Windows、Linux和Mac OS X。



移动开发新宠：Lua语言

当移动开发成为如日中天的技术话题，这必将带动一些新的技术热点的兴起，之前并未广泛应用的Lua语言，因其灵活、轻量、高效的效率成为iSO平台上的新宠。

TIOBE编程语言流行度排行榜的统计数据来源于Google、Bing、Yahoo!等主流搜索引擎，以及各大技术网站发布的招聘信息，在很大程度上代表了编程语言的流行趋势。在本月的TIOBE编程语言排名中，Lua语言第一次进入了排行榜的前十名。

轻量级脚本语言Lua在开发者中正变得越来越流行，这与移动应用开发的升温密不可分。以iPhone为代表的移动平台，为不少开发者提供了新的应用展示空间，也为支持iOS系统的应用开发带来了巨大商机。包括大获成功的游戏《愤怒的小鸟》在内，不少iPhone和iPad流行应用都使用了Lua作为游戏脚本。

Lua语言最大的优势在于其轻量级脚本和高效的运行效率，这一得天独厚的优势让Lua在以娱乐和用户体验而著称的iOS系统之上大有用武之地。

小巧的Lua语言来自于巴西里约热内卢天主教大学于1993年启动的实验性项目。Lua语法简洁优美，解释器由标准C编写。以快速小巧著称的Lua拥有在目前所有脚本引擎中最快的运行速度，这样的特性让Lua成为用作嵌入式脚本的最佳选择。

Lua代码几乎能在所有操作系统和移动终端上编译运行，同时Lua提供了LuaJIT项目（http://luajit.org/），针对特定目标平台为开发者提供即时编译功能，进一步增强Lua的执行效率。

Lua的设计初衷并非意图构建完整的应用，而是嵌入在应用程序中为应用提供灵活的扩展和定制功能，所以Lua仅提供了基本的数学运算和字符串处理等函数库，而并未涵盖程序设计的方方面面。

在语言兼容性上，Lua一开始就瞄准了具有广泛应用场景的C和C++，让C/C++编写的代码可以很方便地调用Lua脚本，同时C/C++函数也能够在Lua脚本中调用。与C/C++代码的良好交互性，让Lua在应用程序开发中有着广泛的应用基础。

Lua最流行的应用场景是在游戏中作为处理逻辑的脚本，即用来编写游戏的任务脚本，以实现游戏的可配置性和可扩展性。在游戏中Lua脚本的加载不需重启服务器，尽管脚本的运行效率低于编译好的二进制代码，但目前CPU速度已完全能够让Lua脚本迅速解释执行。

在以《魔兽世界》为代表的网络游戏中，几乎所有的插件都是使用解释性的脚本Lua编写，核心的C++代码只占全部代码中的很小一部分比重。

目前Lua语言的最新版本是5.2 Beta，如果你也准备在移动应用开发中尝试嵌入Lua脚本，在熟悉语法之后，便可移步Lua开源项目站点luaforge.net或Lua开发者站点lua-users.org的Wiki库，下载感兴趣的Lua程序脚本，开启Lua脚本的学习之旅。



高昂
中国标准化研究院助理研究员，从事信息技术标准化研究工作。关注开源社区，也是OSGeo中国和InfoQ中文站成员。

O2O，看上去很美

随着Groupon模式在全球巨大成功，一个新的名词O2O走入人们的视野，并涌现了众多跟随模仿者。要在这样一种新的模式下掘金，需要解决怎样的问题呢？

电子商务领域，各种时髦概念名词总是层出不穷。继B2B、B2C、C2C之后，O2O（Online-to-Offline）作为一种新诞生的电子商务模式，一夜蹿红并受到各方追捧。

O2O反映了互联网商业模式的一个趋势，就是从线上走

到线下，更强调与现实世界的对接和融合，对线下商务活动的直接影响和作用。O2O让互联网成为线下交易的前台，线下服务可以通过线上营销推广方式（如团购、预定、目录服务等）来获取客户，消费者可以在线上筛选服务，成交和支付也可以在线结算完成。由于每笔交易都已经在网上成交，O2O模式的优势是推广效果可精确量化，每笔交易的用户行为数据被沉淀下来，可供跟踪和下一步的数据挖掘，进行精确营销、交叉营销。Groupon被作为O2O迄今为止最典型的成功案例，而O2O也不仅仅限于本地化的生活消费服务领域，还涌现了Uber、Airbnb、Zaarly等创新应用代表，已经超越了网络世界，切切实实为用户解决在现实世界中的困难和问题。

庄帅认为，O2O不是一个新概念，2006年美国沃尔玛就提出S2S（Site-to-Store）的B2C新战略，通过B2C完成订单的汇总及在线支付，顾客到沃尔玛线下4000多家连锁店取货，从而大幅降低运营、物流成本和POS排队，提升店内关联销售。凭借S2S战略，2008年沃尔玛在线逼近亚马逊成全美第二大B2C。

从S2S概念演化到O2O，发生了两个变化：一是提供给消费者的从可以“装箱子”的商品到需要实地体验和享受的服务，如酒吧、KTV、餐馆、加油站、理发店、健身房、干洗店等，开启了一个“万亿美金”增量市场的想象空间；二是更量化和透明地体现线上环节的价值，更深程度地参与线下商务的利益分配，这一点更为本质。O2O通过服务合约被提前达成交易，使得线上环节在整个商务交易链的价值不再模糊，可以精准、量化地呈现效果，从而理直气壮地拿走更大一块的蛋糕。

正因如此，O2O被媒体称为“下一座金矿”，业界意见领袖如李开复等也在不同场合表示“重点关注”、“非常看好”。且慢欢呼，在掘金之前，O2O模式在中国至少有3大问题需要解决。

首先，中国整个社会商业体系的信息化水平还是太低，O2O模式在完成线上与线下的对接和融合上，需要解决信息基础化设施的问题，例如商家自身的客户关系管理能力的成熟度。国外以SaaS模式服务中小企业的Salesforce的CRM比较普及，相比之下国内这一块还很薄弱。而且，服务类是非标准化，对非标准化的实物产品如家居等电子商业也还在探索之中。

其次，需要一个类似支付宝的对商家的服务质量、售后质量的第三方保证体系。“付款前是上帝，付款后不如小弟”，当消费者预先在线上完成支付购买，实际体验享受服务时的满意度很难保证，有问题之后投诉也很困难，最多在网上发发牢骚。目前屡屡曝光的团购平台、消费者、商家之间纷争的乱象就是典型案例，而O2O模式更存在加剧和放大这种冲突的可能。

最后，商家也需要平衡线上渠道和线下渠道的关系。线上线下价格协调、货品协调、线上线下活动贯通等，这些都是O2O模式要在实践中探索解决的。目前，线上支付对消费者最大的诱惑是低廉的价格，但对普通消费者是一种不公平。最近Groupon准备上市，这个问题也是热议焦点之一。除了所谓“瑜伽教室”这种说辞，商家也要考虑新顾客留存率、对毛利和利润的冲击。即使在比较成熟的B2C领域，线上和线下渠道之间的利益与关系的平衡，也还没有得到很好解决。

对这些问题，SoLoMo是一个技术解决手段，但还远远不够。正如B2B模式在中国，当初面临的物流配送、第三方支付等基础设施问题也是逐渐在发展中解决的。团购、短距电子商务（如“快书包”）、SaaS的发展，会逐渐改善O2O的大环境，创业者也可以找到类似携程当年“发卡片”的创造性的具有中国特色的变通解决办法。



高巍

高巍，安卓爱普公司创始人。原搜狐媒体产品中心经理。关注移动互联网、互联网产品管理。

微博：httpt.sina.com.cn/inetpm

一个世界，一个梦想

普通玩家可能并不知道，自己面对的游戏世界是多方妥协的产物，同时也是不同游戏制作人员殚精竭虑的产物，如何构造奇妙多样、引人入胜的游戏世界呢？这个问题的确值得认真考虑。

从事游戏开发已经超过10个春秋，除了职业生涯一开始的两三年，基本都是在和网络游戏开发较劲。开发网络游戏给我带来了很大的乐趣，因为它是一个很有创造性表现的舞台，在游戏中，你可以模拟世界上的事物，也可以创造一些

这个世界上不存在的东西。你就是这个世界的创世神，这个感觉很美妙!

但是，在真正具体的项目研发中，团队往往都会受制于资金、技术实力、档期甚至中国玩家的习惯等一些因素的影响，通常最后呈现在玩家面前的游戏世界会是各方妥协的产物。对于一个虚拟的网络游戏世界，我有一个梦……

让我们想象一下，在一个广袤无边的世界，自然地分布着山川、河流、湖泊、海洋，各种原生态的生物在这里繁衍生息，甚至进化。

玩家来到这个世界，组成不同的种群部落，一点点改造这个世界，通过玩家们得努力建设，世界上出现各种各样的村庄、城镇、甚至王国。玩家可以挖开山体，可以在河道上架设桥梁。人们成群结队的外出捕猎野兽，获得各种材料。而野兽们则符合生命基因传承的不断进化……是不是有些像单机游戏《孢子》？

对于这样一个游戏世界，不同于目前的网络游戏中玩家按照开发者规定的路线，按部就班地发展，而是一切都是可以改变的，玩家的任何行为，都有可能导致世界的变化。听起来太美妙了，我相信有这个梦想的绝对不止我一个。但是要实现他们，是多么的不易，遇到的阻力是难以想象的。我这里想要以一个程序员的角度，尝试去思考那种可能，这是一个很大的命题，也许我终其一生也不得要领。但谁没有一个梦呢?

要构筑这么一个游戏世界，首先，我们要考虑的问题就是如何制作巨大的世界地图问题。我们先从这里开始吧。超大世界地图，这对于目前游戏开发技术来说，已经成为了现实，wow的连续大世界就已经非常成熟稳定了，就算我本人在2003年底，也成功实现了一套超大世界的漫游系统应用在一个已经消失在历史长河的项目中。

但是，这样的地图满足我梦想中的世界了吗？显然没有。首先，它还不够大，另外一个最重要的事情，这个世界是一个死的世界。为什么说是一个死的世界呢？因为哪怕我想在这个世界永远挖个坑，都是不被系统允许的。

其实，要让一个地图更大，技术上也不是什么特别的难事，但是当地图大到一定的地步，这个世界的原始地貌如果让关卡编辑人员来做，那就会是一个让人崩溃的课题。下面，我现在试图开始来解决这个问题：

超大世界，目前成熟的解决方案，大体从程序逻辑上说，都是把我们要做的世界分解成一块一块的子地图。然后通过levelstreaming技术，多线程后台读取热点地图，释放不关心的老地图，这样达到理论无限的世界地图。这一点很好，我们要继续保持，但是这个巨大的世界我们需要赋予一定的初始状态，在地图大到人力不足以完成具体编辑设计的时候，我的想法是可以考虑采用整体规划加分形来自动产生世界地图!

所谓整体规划，我想就是可以给予这整个地图一个大致的地势走势分布、植被分布、水源分布。地图初始化系统会根据这些信息自动给予一些最初的地图区域属性。经过这一步后，再利用一些随机因子来扰动这些属性到细节的地形信息。

目前，互联网上，关于分形算法和自动生成符合自然特征地形地貌的技术论文和demo也有不少，做一定处理就可以产生出比较合理漂亮的地形地貌。需要注意的是因为超大世界的特性，我们不可能一次在内存内把这些处理都做完，分子地图逐渐处理是必然的结果，所以子地图之间的接缝处理需要引起重视。

对于这个部分，我和我朋友做了一定的技术常识，目前已经能产生出比较理想的超大地形世界了，相信随着各方面技术的进步和积累，人们一定会在不久的将来跨越这个问题。

上面还说到一个问题，我梦想中的世界，是一个多变的世界。玩家能真正改变世界，比如人们可以开山、搭桥、砍伐树木、建设城堡等。那么技术上，我们是否能做到呢？对此，我有一个不算成熟的想法，就是在游戏内实现数据，控制的分离。

游戏内，最初的数据一旦确定了，后面的一切都围绕如何修改数据的操作来做文章。改变数据的操作，在我的体系里面被称为修改者（Modifier）。当对象数据被多个Modifier修改后，最后会输出称为游戏内最重看到用到的对象。这样，在网络传输中，我们避免了数据本身的传输，一旦出现数据变更修改，我们只需要把modifier同步给相关玩家就可以了，设计合理的modifier可以大大降低网络传输的压力，从而使得玩家个体行为改变整个世界成为了可能。

说到我这个梦想，就像打开了一个巨大水库的大坝，各种问题如奔腾巨浪碾压过来，让人心存畏惧。也许，这是一个我永远都实现不了的梦想，但回到我一开始的那句话，谁没有梦想呢？与诸位朋友共勉了。



宋忆疆

参与《碧血情天》、《傲世三国2》、《乱舞天下》等游戏研发。目前担任《流星蝴蝶剑OL》项目制作人。

商务智能山楂味

商务智能有味道吗？是否食之无味，弃之可惜；抑或是彼之砒霜，吾之蜜糖。商务智能这个大盘子里越装越多，分一杯羹的人也越来越多，谁又尝出它真正是味道呢？

作为一个快速膨胀的市场，用户对于商务智能系统从“相知、相识”的卖方市场已经慢慢过渡到“挑肥拣瘦”的买方市场。

年初，Gartner的一则调差显示70%~80%的商务智能项目以失败告终，调查一出即在业内造成一场讨论，其中包括对各种导致失败因素的分析，甚至就连怎么认定一个商务智能成功的标准也争得不亦乐乎。不过，本月一家知名在线商务智能服务提供商LogiXML发布的调查报告再次显示令人失望的结果——半数用户不满意。

业内出现了一个相悖的现象：一方面大批商务智能项目的投资不断打水漂，用户花大价钱“请神容易送神难”，把自己折磨得痛苦难堪；另一方面45%的用户还表示明年将继续加大商务智能投资，纷纷掏出钞票推动商务智能产业“大跃进”。

为什么会出现这种局面?

先说后一个，主要原因在于用户的商业压力，甚至仅仅是投资部门希望通过商务智能项目向企业高层“邀功”、增加高层对部门的印象分。而造成前一现象原因似乎更复杂，以前一般认为在于业务、技术双方的衔接问题，当然这个很重要，但调查同样显示出另两个问题。

问题一：实施商务智能项目的团队没有足够的技术力量解决用户的需求。

问题二：一旦项目移交给用户，用户也无法在预算允许的范围内获得必要的技术资源去运营其商务智能项目。

首先，你可能认为问题一不是问题，毕竟现在市场上商务智能产品、解决方案供应商这么多，怎还做不了这类项目？用户抱怨开发团队与他们沟通困难，因为开发团队似乎总在试图将很多他们不熟悉、不好理解的调查问卷“塞给”他们，在业务需求调研中总是问不出“好”问题，而且开发团队似乎也没想真正弄懂用户业务上到底要做些什么。

项目实施过程中，开发团队拖沓，交付时间又总是一拖再拖，等到开发接近尾声，丑媳妇终于要见公婆的时候才发现原来在错误的道路上已经“南辕北辙”了很久。不仅如此，似乎商务智能项目的投资就是无底洞，为了能够按时获得一张报表，用户只好源源不断地投资硬件、购买产品，但过一段时间“慢”又成了主旋律，面对这些问题开发团队似乎也总是一筹莫展、江郎才尽。尽管作为开发人员，我们可以找到各种客观原因说明这是客户的问题，但能否站在用户角度反过来想这个问题，也许我们准备得也不充分。

其次，你可能认为第二个问题也不是问题，但请注意一个前提“在预算允许的范围内”，尽管能提供商务智能运维服务的公司很多，但这类项目“看病难、看病贵”的问题也很明显：一方面调查显示客户95%的商务智能项目是“多国部队”，各种产品大杂烩；另一方面因为商务智能项目为了更好的服务客户需要，多多少少都有定制开发工作。

因此维护这些项目从卖家看自然要开个好价，而用户似乎并没做好预算，甚至可以说他们套用以往的维保经验计算商务智能维保费用所用的公式本身就是错的。商务智能项目有时候上得起、用不起。

既然商务智能项目客户觉得不满意、开发团队也一肚子苦水，但为什么这个市场却在“大跃进”呢?

关键在于用户确实需要，前几年的SOA、这两年的物联网、云计算对于用户而言很多东西离他们太远，而商务智能不同，处理的就是身边的事情，显示的也是亟需的各种数据。

所以，尽管前面自己或者同行70%~80%的项目失败，但用户还是愿意继续投资而且是加大这部分投资。

商务智能，想说爱你很难，但偏偏又总欲罢不能。



王翔
软件架构师，主要研究方向为XML、.NET、领域设计和PKI应用。
工作之余喜爱旅游、写作和烹饪。

走在开放平台的康庄道上

现在各方纷纷吆喝着要开放，主办方笑脸迎客，第三方应用摩拳擦掌。这是否一条康庄大道呢？还是说再宽的马路也闹不过车多。

360与QQ大战后，国内开放平台今年异乎寻常地火爆起来。所谓开放平台（Open Platform），就是首先提供一个基本的服务，然后通过开放自身的接口，使得第三方开发者得以通过运用和组装其接口以及其他第三方服务接口产生新的应用，并且使得该应用能够统一运行在这个平台之上，我们把这样的一种网络服务模式叫做开放平台。

开放平台模式成功的要点在于，通过自身服务和第三方应用的互利互惠，提高用户对平台网站的黏性和使用程度，进而提高获利。

同时，通过利益分摊，达到平台自身和第三方应用循环刺激而产生的滚雪球式的增长（资料来源：《程序员》杂志2008年7月刊“OpenAPI出现、起源与现状”）。

开放平台这个牛哄哄的名词，最初由Facebook提出。2007年5月，Facebook正式开放其应用编程接口（Application Programming Interface），这被称为Facebook开放平台（Facebook Platform），第三方软件开发者可以开发在Facebook网站运行的应用程序。

不久Google也宣布开放自己的社交应用平台OpenSocial，随后，微软、Flickr、YouTube等一系列公司和网站也都紧随其后，纷纷对自己的服务提供了API供第三方开发者使用。

一时间，“开放平台”成了全球互联网中最为流行的关键词之一。这股风没隔多久，也被国内的互联网企业迅速抓住。淘宝、百度、腾讯、360、百度、搜狐、开心网、人人网等纷纷推出自己的开放平台战略。

据我观察，国外开放平台的秘密在于应用，国内开放平台的秘密则在于流量。这是国情不同造成的。国外你只要做潜心做很好的应用，做一款爆款软件发财也不是不可能。

但在国内，第三方应用，往往并没有把心思放在自己的产品上，相当一部分都放在了如何从别家平台上招商分流为我所用上。

天下没有免费的午餐，就在百度处心积虑构建自己的通用搜索局域网，淘宝千方百计构建购物搜索局域网，腾讯事实早已经很多年垄断IM市场的时候，他们几乎为什么同时选择开放了呢?

中国互联网产业的见证者李开复认为，进入互联网时代，平台更重要，而且如果不开放，就会被更开放的平台取代。“七年以前，Facebook是一个小小的网站，在2007年，Facebook做了一个重大的，可能也是艰难的决定，就是把它的平台开放出来，从2007年以后，Facebook超过了MySpace”。

创业者有分工，越开放，对生态系统越有益。李开复同时认为，互联网创业者黄金时代来临，创业要赶紧。

其实，老邢我关注淘宝和新浪微博开放平台很久了。两年前，我就尝试在淘宝开放平台上做一些尝试性的开发，直到2010年年底，则全职投身到淘宝API开放平台上，并结合自己十几年的开发经验，这个月终于在淘宝上成功上线了自己的进销存软件应用。

期间我还在电子工业出版社出版了一本淘宝API开放平台开发经验的图书。我这半年多的感觉就是，草根创业者，创业一定要站在巨人的肩膀上，就像李开复所说，由于大佬们纷纷开放了自己的平台，互联网创业者黄金时代再一次来临了。

如果说1999~2000年的第一次互联网泡沫是张朝阳海归们的机会，这一次则是草根们站在他们的肩膀上，再次创业的好机会。

但这条道路并不好走，因为开放的最终解释权，都掌握在百度、360、淘宝、腾讯这些平台缔造者手中，即使大家都开放了，马化腾甚至还发毒誓，腾讯全开放平台绝不向360开放。

什么是开放？你的开放跟它的开放压根接不上，这能叫真正的开放吗?

这类潜规则，在Twitter也一样体现出来了，Twitter不仅取缔了一些与Twitter官方应用程序重复的第三方客户端程序，还对第三方应用访问权限进行严格限制。套用我一个朋友的话：绝对公平是绝对不存在的，即使是在开放平台的名义下。

但开放平台的前途无疑是光明的，创业者再想通过一个

idea，开创一个属于自己的平台，成功的概率可以说跟中大奖差不多。开放平台上的道路也不是想象当中的那么宽广的。所以，我说让我们走在开放平台的康庄小道上。话说回来，目前虽然窄点，但车少。就像北京的大马路，终究会越来越宽的，潜规则也会越来越少的，但开在上面的汽车，以后也会爆炸性增长起来的。



邢波涛
北京新软孚信息技术有限公司技术负责人。关注SaaS管理软件和B2B、B2C电子商务的融合。

覆巢时代

黑客们开始针对基础设施进行攻击，这一举动造成的影响，使国事民生的金融、工业、军事等领域的安全岌岌可危，也许我们需要确保的是在覆巢之下，做一颗不那么容易碎掉的卵。

本月最令人震惊的不是洛克希德马丁、诺思罗普格鲁曼这些军火巨头厂商遭受入侵，而是这些入侵具有共同的原点：RSA的SecurID令牌。

这几起令人震惊的入侵事件，都源于令牌仿冒，而仿冒的原因经推导是今年三月RSA所遭遇的入侵。基本可以确认与用户账户关联的伪随机数生成算法或其种子遭到了窃取。因此，RSA已经发放的所有令牌从某种意义上已经失去了安全价值，而成为一颗随时可能爆炸的地雷。

目前，RSA已经宣布召回大约4000万SecurID令牌。这是硬件安全性神话的再一次破灭。

一段时间以来，特定的攻击正在转向针对基础设施，而基础设施的重要环节就是传统的电子认证和信用体系。因为更多的安全应用基于此搭建，从去年VeriSign遭遇入侵的传闻，到今年3月RSA和Comodo（SSL证书厂商）的沦陷，其连锁反应都还在持续进行当中。联想到这些厂商的产品在金融、工业、军事等领域的广泛应用，一个“覆巢之下岂有完卵”的时代正在到来。

而在覆巢时代，带来更大变局的就是国家力量的身影。从Stuxnet蠕虫被怀疑是以色列情报部门所为，到大国之间频繁的相互指斥，都可以嗅到浓重的硝烟味。

今年五月，美国部分网络安全专业人士向美国政府就利比亚战事呈文，这篇名为Project Cyber Dawn Public的文章不慎在网络上泄露。发布者在经过关键字涂抹后，干脆将其完全公开，以冲销流失的影响。

这是一份非常具有典范意义的评估报告，从一定意义上，这也是一份信息战的预案和前言。

相比这下，此前被视为臭名昭著的黑客组织“lulz”，本月显得可爱得多。在游戏机厂商世嘉被入侵后，他们声称希望为世嘉提供技术支持，并代为报复。

在索尼、任天堂先后被攻破后，世嘉出问题也就成了早晚的事情。但这些入侵，都以用户信息库为目的，用户在紧盯自己的电脑的同时，别忘了我们的隐私也在“云端”。

笔者本月提出一个观点：传统的HASH算法不再适合用于密码保存。HASH的初衷并非是用于口令保护，而是用于完整性验证，口令由于受其长度限制，是一个有限集。在GPU等辅助计算资源无比强大的情况下，廉价的计算中心正在不断积累覆盖这个有限范围的HASH值。

一旦口令数据库被攻破，HASH的算法单向性已经不足以保证口令不被还原。而多数用户基本上是在很多系统中采用同一口令，攻击者获得大批用户信息和口令后，和用户信息关联，再去攻击用户的其他系统。

这种攻击方法目前已经成为了一个非常显著的关联威胁。因此，在口令保护中，设计新的、带有环境参数机制的专用HASH算法很有必要，这样同一个口令，在不同系统中的HASH结果均会不同，查表法就会大打折扣。



肖新光
网名江海客，安天实验室首席技术架构师，研究方向为反病毒和计算机犯罪取证等。

探访硅谷创新企业：软件工程师的乐园

特邀记者 / 巩文坚

美国时间6月6日~10日，在苹果开发者大会召开期间，本刊特邀记者探访了数家硅谷创新企业，其中包括大家耳熟能详的Facebook、Google及Github，同时还参观了美国计算机科学博物馆。我们就其中精彩部分做图片报道，带大家一起去领略IT领袖企业的风彩。



这是Facebook位于加州Palo Alto的总部，注意看下图中大门上的文字，看到了吗？HACK，对，就是hack，这也是Facebook独有的hack文化，意思是要求员工不要受各种约束地思考创新。

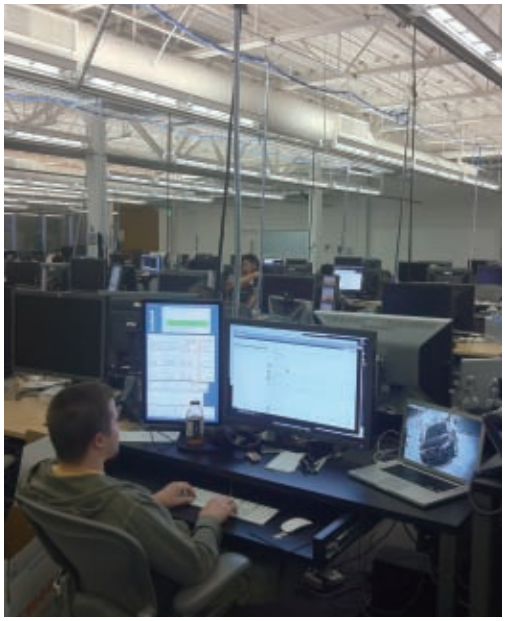


这是Facebook总部大门口的涂鸦墙，里面密密麻麻地写满了参观者的签名，看到那个大大的“拆”字了吗？没错，Facebook不久就要搬离这里了——扩展太快，地方不够了。



这是Facebook高层领导的办公桌。左边是Facebook CEO马克·扎克伯格的办公桌，右边是CTO的，他们和所有其他员工一样坐在开放的办公室，享用同样大小的空间。扎克伯格的桌子上放着一本基辛格的新书《论中国》，看起来他们对如何进入中国，确实用心思虑。

尽管已经是家大公司，Facebook仍然保留着创业公司的特质，在偌大的厂房式的建筑里没有任何装修，各种连线直接从天花板上垂落，远远望去，颇为壮观。





这里是Android的大本营，Google公司内负责Android业务研发、推广业务群组都集中在这栋楼里。Andy Rubin也在这里哟，Android控们赶快过来朝圣吧。



Android各版本都用甜点做代号，如Froyo2.2、Gingerbread 2.3、Honeycomb3.0、Ice Cream Sandwich，你找到它们了吗？



Google，尽管已经是世界顶级公司，它的办公楼群却不像同在硅谷的其他大公司如思科、苹果们一样有着统一的建筑和装饰风格，这是由于Google扩展速度太快，来不及自己建楼，全都是租买而来，这个园区原来是做图形图像处理的顶级企业SGI的，后来SGI衰败，归了Google。

Github公司位于旧金山拥挤的城区，在寸土寸金的区域中一个不起眼的小楼安营扎寨。他们的办公室更像是个厂房，但粗陋的红墙、简约的装修却挡不住业绩翻跟斗似地增长，马上，他们也要搬家了。





加州的计算机历史博物馆陈列着各种在计算机发展史上起了很大作用的机器。此图是由Steve Wozniak设计并手工打造的Apple-I电脑，是苹果历史上的首款产品。



此图是苹果历史上传奇般的神机——Apple-II，它是苹果公司首个普及型产品，是Apple-I的直系后代，曾风靡一时，影响了一代电脑人。



计算机发展史上也留下了华人的身影，王安公司的文字处理机曾是美国各公司的必备，此图就是王安的代表作2200文字处理机。



计算机历史博物馆出口挂着这样的标语“Today is so yesterday”。计算机与互联网飞速发展，众多名企如过眼烟云成为历史。俱往矣，数风流人物，还看今朝！

硅谷直通车——离岸软件产品研发

记者 / 董世晓

随着国际软件巨头纷纷创建或者扩建其在华研发基地，比如微软刚刚在中关村核心区域落成了研发大楼，意味着越来越多的软件企业将中国的软件研发实力，作为今后可持续性发展的动力。

面对如此蓬勃的发展趋势，此前一直为外界所忽视的离岸软件产品研发，越来越受到行业的瞩目，并得到了良性的发展。作为离岸产品研发的代表企业，刚刚迎来公司创建十周年华诞的奥博杰天近日更是隆重宣布将其设在西安软件园的分公司升级为具有独立法人资格的、功能齐全的子公司，颇有一番蓄势待发之势，这也从侧面反映了离岸产品研发形势喜人。那到底什么是离岸产品研发？它有哪些优势？还存在哪些瓶颈？带着这些疑问，本文将为您一探究竟。

何谓离岸产品研发

与传统的IT外包不同，离岸产品研发涵盖了软件产品从设计、开发到支持的整个产品开发生命周期。可以说，离岸产品研发企业，会掌握需方软件产品的核心技术、设计及支持。

提起产品研发，公众的普遍认识是“类似微软、Oracle、IBM、EMC、SAP等软件巨擎都是在其总部的研发基地完成产品的设计、开发、测试等环节，在各个国家，大多是通过其软件合作伙伴，为终端客户提供个性化的订制和咨询服务”，如此认识也导致公众对于离岸产品研发存在一定误差，即发展中国家如中国，是否有足够的软件技术人才和实力，帮助企业实现软件研发的全球化分工。

目前，中国的软件人才和研发实力正在受到全世界的青睐，微软亚太研发集团/亚洲工程院、IBM中国研究院等国际软件巨头在华扩研发基地，承担了其大量核心产品的研发；同时，对于它们的主要客户，这些研发基地也负责为其客户提供有关的研发服务，但是价格高昂。对于那些青睐中国的软件研发实力，但又支付不起高昂费用的跨国企业，特别是大量的独立软件开发商（ISV）来说，离岸研发基地能很好地弥补这一空白。

在竞争激烈的软件行业中，离岸产品研发是一个不易被察觉的价值洼地。由于不是简单地参与做些边边角角的工作，而是深入软件产品的完整生命周期，所以就像开发自己的软件一样，拥有较大的主动权，并能投入极大的热情。而且，需方能将软件需求发出来，肯定也是经过缜密的市场调研，说明了该软件存在不菲的价值，并且与其关键业务密切相关。为了在瞬息万变的竞争中把握先机，需方必然会对其迫切需要的软件，求之若渴，也自然就舍得花大价钱找离岸研发企业帮忙。

正如奥博杰天总经理谈益平博士所描述的那样：“我们独特的离岸产品研发模式，力求在中国的软件研发实力和快速多变的市场环境之间寻求一种突破，即：大量ISV需要应对市场的需求提供基于新技术、新应用、新理念的产品，而又很难在短时间内组建一支召之即来，来之则战的精干研发团队。”我们不难看出这种“不走寻常路”的软件企业发展模式，正在成为一种潜在的动力，引导中国的软件企业，从“红海”走向更加宽广的“蓝海”。

不出国门，学习先进技术和知识

离岸软件研发一般都是直接面向欧美企业，而它们对软件产品的需求基本反映了国外先进技术的现状，所以离岸研发企业会在软件研发的过程中，得以掌握和使用最新、最先进的技术和平台。借助这个反映世界先进技术潮流的平台，比如亚马逊云、微软云等，让中国员工有机会接触国外先进理念和技术，跳出固有思维的束缚，开拓思路，进行创新。

由于在研发的过程中，客户的需求、产品、支持等相关部门都要大量的参与讨论，因此对于供需双方的沟通能力和技巧，都有很高的门槛。在沟通的过程中，既需要离岸研发企业提供优秀的软件工程师和强大的研发实力，也需要从研发的角度提供很好的反馈意见。往往经过这样反复的过程，供需双方形成了长久和稳固的合作关系，这可以从目前奥博杰天整个公司的组织架构都是以一个个主要客户的研发团队为基础组成的，可见端倪；同时离岸研发企业的工程师队伍，也由于经受了从技术到文化的融合，再到沟通能力的综合锻炼，反而在就业市场中变得炙手可热。

人才是离岸产品研发的最大瓶颈

不可否认，由于公众对软件研发的思维定势，很多人对离岸软件研发并不感冒，导致该行业的人才需求缺口极大，尤以具备英语沟通能力的项目管理人才为甚。但是随着众多软件巨头纷纷在华扩建其研发基地，有的甚至将原来设在印度等地的研发基地转移到中国来，我们不难看出，中国产品研发实力正在由传统的“中国制造”向“中国创造”，进行历史性的跨越。在这样的历史性机遇到来之前，离岸产品研发自然也摆脱不了人才稀缺的干系，并成为制约其发展的一大瓶颈。要消除这个瓶颈，一方面需要继续加大人才的培养力度，另一方面也是更重要的是要从理念上进行引导，提升公众对离岸研发的认识。

此外，中国的软件行业一直在模仿的印度模式目前遇到了一个比较大的问题——由于竞争越来越激烈，从而导致人员流失严重，每年的人员流失率达到30~40%，致使知识无法健康传承。而以离岸产品研发为基础形成的人员管理模式，却往往由于离岸产品研发的高稳定性、技术能力的高可用性、以及人员对于西方文化的融合性，反而能够在竞争激烈的环境中，独树一帜。



5月31日，离岸软件产品研发企业奥博杰天的西安分公司成立

“我们的客户往往是冲着我们的研发实力来的，我们的合作是长久和稳固的，在合作过程中，参与的工程师通过不断地接触新知识、新理念、新技术，从而形成了高超的技术实力、敏锐的‘第六感’和良好的沟通技巧，更提升了他们的独立思考的能力，这往往成为海外ISV选择我们的主要依据，即：我们的工程师并不只是按部就班地完成需方的工作，反过来我们积极地参与和讨论，往往能够给需方带来新的‘活水’。相信我们十年来摸索出的‘人尽其用’的用人理念，并配合国家及各级领导机构的政策导向，能够很好地解决人才瓶颈这一关键问题”，奥博杰天总经理谈益平博士如是说。

离岸产品研发的未来

人的一生充满了梦想和挑战，并在不断的磨炼中实现自己的价值。当年，年轻的Bill Gates正是从一个有着独立思考的程序员角度，本着“让所有人的桌面上都有一台计算机”的梦想，通过一款软件创建了今天的微软帝国。在现今的数字化革命洪流之中，众多有理想、热情、抱负的离岸产品研发工程师，正在帮助越来越多的企业实现其愿景，这也是离岸产品研发的未来。





寻找技术领袖

策划 / 本刊编辑部

作为智力密集型行业，无论大小，每一个IT企业都会有自己的技术负责人。他们决策着公司的技术选型/架构，技术团队组织等等，这很大程度上，决定了企业的根基是否牢固、未来发展是否顺畅，并且他们的决策将形成一股合力，决定了时代的技术发展与潮流。这其中，总有几位特别耀眼的决策者，较之普通的CTO/CIO，他们对技术的思考更深入、更广泛，对技术发展、技术团队模式的形成，推动力更大；他们对技术的见解、言行备受重视；他们是众多技术开发人员眼中值得敬重的前辈，是公司可堪信任的技术大牛；他们是这个时代的技术领导人。

技术领袖的共性

至此遇到的问题是，怎么样的人可称得上技术领袖？我们发现这些在技术领域取得突出成就的人存在着若干共性。

首先是懂技术，虽然很多人认为技术领袖身上更应该体现出一种高屋建瓴的远见，但在他们的特质里，技术永远摆在第一位。

如果说一个公司的技术领导者，是用技术服务所在公司的商业模式，那么技术领袖在此同时，还用技术服务着整个行业。

从成长经历看，虽然接触行业的时间有早晚，但他们无疑都是人们眼中的“技术天才”，对新生技术敏感性极高，对每一种新推出的技术，都有着比较深入的研究和了解。良好的技术素养，使得他们不需要深入技术细节，在了解大致背景的情况下，也能保证对产品发展、升级方向有个把握。如此一来，他们总能

第一时间把握到技术发展的走向，甚至借势作出令人惊艳的产品。

虽然有些人在刚开始从事技术管理时，有“技术味儿”过重的阶段，但公司的创新产品、或者经历重大技术转型，都是在他们的带领下完成的。

其次有着敏锐的商业嗅觉，技术领袖在了解过往技术发展的同时，也要有对未来技术潮流的把握，，并且有着出色分析能力。这样面对层出不穷、眼花缭乱的新技术时，才能有所凭藉、有所判断。在面对复杂、存在风险的项目时，能够详细分析，从项目影响力、开发难度、最终收益、后续维护成本，可借鉴性、客户需求变更等多个个方便进行评估，然后综合以上情况，对整个项目做取舍。整合公司的商业能力，对公司核心技术和核心业务有着非同常人的理解，这些都是一名技术领导人必备的要素。

再次，对技术人才的理解非常深刻。因为软件开发的特殊性，增加人手、延长开发时间往往是徒劳无功的。最为有效的方法是网罗合理的技术人才，而这要付出很大的投入，冒很大的风险，以至于技 术人才的选拔与甄别在技术领导人日常工作中的权重非常大。在获取优秀人才方面，他们的经验无疑值得借鉴，他们有的注重打造公司品牌、完善从底层培养人才的机制；有的保持和高校、社区的密切联系，孜孜不倦地参与各种技术圈子的建成与发展。这些都成为吸引人才的源头活水，他们就像一个熟练的果园主，培养好苗子的同时，自己也取得了丰收。

还有注重技术团队的管理。技术领袖对于人才招募，组建团队，都是亲历亲为，即使公司结构、人事经历调整，也能保持公司的技术底蕴和文化氛围的基因不致失真。关于技术团队的管理，他们都深知技术人才是无法量化的，只有做到知人善任，并都信任懂技术的“内行人”，才能处理好技术与管理的平衡关系，保持团队的继续发展。

最后，堪称技术领袖的人无疑是有个人魅力，并且对成功有着不懈的努力。

这些优势的共性，常常重叠，就像同一醒目建筑物不同方向上的不同窗户，每一扇窗和都展现出自身的特征，都能从它们周围的窗户上看到。

对技术领袖认知的改变

对技术领袖认知的改变出现在两方面。

一、科技的普及，技术门槛降低，使得更多人参与到技术管理工作中，在大家都对成为领导者加大投入的情况下，技术领导技术人要想脱颖而出，成为一呼百应的技术领袖，越来越困难。但正如Weinberg在《技术领导之路》一书中强调的，我们身上都蕴藏有领导力。只是对不同的人来说，某方面的因素发展得比其他方面的因素更好点。他写道：“只要加强自己最弱的元素。任何人都能成为优秀的领导。健美冠军的肌肉不比我的多，只是锻炼得更好而已。”

二、产品用户的要求多样化了，单一技术/产品根本无法满足用户的需求，只能靠众多富有个性的公司推出多样化的产品，这其中有着越来越多的小团队甚至一人团队出现，这将使得技术核心人物的作用愈发重要，但这对技术领袖综合能力的要求也会越来越高。

这两方面的改变，都在标示着一个信号，这个时代的技术领袖更加弥少而愈珍，这也是对他们了解的价值所在。

结语

本次封面报道，名为“寻找技术领袖”，在面对面的采访中，我们们向八位技术人发问、探讨、了解，希望能够展现在光鲜的产品、枯燥的技术背后，有着怎样的给人启迪、发人感慨的故事，找到塑造这个时代技术潮流的领导气质，并且记录他们的成长、成功以及感悟，分享给广大技术人员，让“每一位想当将军的士兵”，都能找到自己的鉴借。

SAP中国研究院总裁芮祥麟

从千里马到伯乐

文 / 芮祥麟

芮祥麟，现任SAP中国研究院总裁，致力于作为SAP全球研发中心之一的中国研究院的建设与发展。在本文中，他将分享自己从求学到工作、从IT技术实践者到管理者转变的经历和思考。

每个人都有自己与众不同的成长道路。当然，我也不例外。

我一直感觉自己很幸运，在求学、工作上都能够顺风顺水。当然，世上没有无根之水，一方面是遇到了赏识自己的伯乐，另一方面更是得益于自身的努力和准备，当机遇来临的时候能够把握住。



SAP中国研究院总裁芮祥麟

玩转微处理器——编程初体验

在我那个年代，很多人儿时的梦想是做火车司机，而我从对职业有概念时起，就立志成为一名技术工程师。由于从小数理化成绩就不错，所以我的升学很顺利，读了台湾最好的初中、最好的高中、最好的大学，甚至大学里最好的系。在求学过程中，除了对理工科的偏爱，我也对语言文史方面拥有浓厚的兴趣，并且自认为很有天分。但后来屈从于“学文史的出来在社会上不好找工作”的现实，我最终放弃了文科，专攻工科。

早在1978年刚考上大学时，我就接触了编程，不过是硬件编程。那时有个儿时的同伴（现在微软总部担任高级主管），拿着一个很奇怪的东西来找我玩——后来知道是电路板。我当时对那块电路板很感兴趣，经过仔细观察后发现，它上面有很多集成电路，是一个早期的微处理器。那时连汇编语言都没有，我的编程从最基础的东西开始慢慢建立，所有指令都要一个Bit一个Bit地输进去，而且数据和程序都在一起。即使是这样，也完全阻挡不了我对这块微处理器的迷恋，对它的爱不释手。

一晃大学时光就过去了，我开始去服兵役。那时计算机这门学科刚刚开始，在台湾没有几个人懂得如何操作电脑，更何况编程，因此我一当兵就被“抓”去编程。编程用的是Basic，而之前我对Basic完全不懂，但没办法，只有我会玩计算机。

服兵役那两年我过得很快乐，当别人扛枪、扛炮的时候，我整天在机房编程，心里难免有点小得意。这期间，我学到了很多，除了硬件，软件也开始入门。因为那时还没有任何所谓的操作系统，要想与别的电脑通过网络沟通的话，需要自己把网线连起来，从头开始建立整个通信协议，要知道，握手协议是20世纪

80年代末期才有的。此外，那个时期有很多数据需要存储，但没有什么数据库，必须用很原始的方法实现读写。

机会青睐有准备的人——德国深造

服完兵役之后，我选择了去德国留学。之所以选择德国作为留学的目的地，也可以说是一种必然。

我父亲是大学教授，他的很多同仁都是从德国留学回来的，给我讲了很多德国的事情。从他们的讲述我感受到，德国是一个强调实践的国家，而我天生属于喜欢动手实践的人，这与德国文化不谋而合，因此从高中时就一直很向往去德国。以至于高考完的暑假里，还没进入大学时我就开始学习德文。

在留学过程中，我最大的收获是学到了一种思维模式——遇到问题，可以建立一套体系架构和流程，找出问题的关键点，然后寻求方法将问题一步步解决。这种思维模式直到现在仍然让我受益匪浅，在遇到问题时总能游刃有余地解决。与台湾只重视知识本身传递的填鸭式教育不同，德国或者西方教育体系最精华的部分是告诉你知识该如何使用。因为知识，尤其IT知识，会淘汰得很快，仅靠上课时肤浅的认知是不行的，学会了运用的方法才能真的让知识成为你的。与此同时，我也开始慢慢接受比较系统的软件教育，比如数据库、数据结构、通信协议、面向对象设计等。因为我的专业是自动控制，一方面需要具备扎实的理论基础——自动控制说到底就是对各种数学模型的研究，另一方面需要具备良好的设计能力，所以这为我在理论和实践方面奠定了坚实的基础。坦白说，对于技术，我一直有强烈的求知欲，因为我始终相信机会青睐的是有准备的人，你有了充足的准备，机遇才会想到你。而正是由于我在理论和实践等方面的积累，为我日后在SAP的顺利发展发挥了重要作用。

小试牛刀——前两份工作

硕士毕业以后，我先是到ThyssenKrupp钢铁公司工业电子部门从事工业控制系统方面的工作。在那里，我学到了德国制造业的整个管理方式，以及制造业如何借助信息技术实现自动化。德国制造业自动化在20世纪80年代中期就已经开始，这也是其生产力一直稳居欧洲冠军的原因。

两年半后，我到了欧洲航空防卫及太空公司，接触到了更先进的信息化实践。20世纪90年代正是整个欧洲特别是德国制造业信息化突飞猛进的阶段，而且欧洲航空防卫及太空公司是欧洲国家联合成立的一家最大的航空、太空公司，欧洲最优秀的工程师都渴望到那儿工



1994年，在北京举办的一次展会上，SAP与IBM联合展示。当时我第一次来到中国，是“中国计划”的先驱部队成员之一，去实现由“德国制造”向“中国制造”转变的梦想。（左起第六位）



2003年11月20日，SAP中国研究院正式成立。当时，我与60位SAP中国研究院的骨干一同继续追寻我们新的梦想：将“中国制造”变为“中国创造”。2006年SAP中国研究院一期正式投入使用，在我身后，是当时在建的研究院二期和三期办公楼。

作，就像渴望到美国波音公司或者到洛克希德·马丁公司工作一样。我在那里受益良多，但越来越感觉到自己的兴趣在向软件方面发展，况且IT技术对甲方来说只是一项支持技术，达不到专精，所以就希望能够到一家真正的IT供应商打拼，因此萌生去意。

最正确的选择——加入SAP

既然去意已决，我便又开始寻找我的下一站。摆在我面前有两种选择：一是到西门子或者BOSCH这种已经成熟的大公司；二是到尚处于发展中的企业SAP。

我的家人和朋友们都很保守，他们一致建议我去西门子或者BOSCH等位居德国蓝筹股的百年老店，不要考虑SAP这种小公司。确实，当时SAP只有600人，是一家名副其实的小公司，在德国的软件行业里，前二十名都排不上。我一直相当犹豫，所以迟迟没有答复SAP的邀约。两个礼拜后，SAP人事部门打电话问我为什么还不去报到，我回复说我还没有想好该不该去报到。又过了两天，SAP创始人之一Hasso Plattner亲自打电话来说：“祥麟，你干嘛不来？”他在SAP的地位有点像Apple的乔布斯，当时主管技术和研发部门，目前是SAP监事会主席。他劝我道：“你应该加入我们，因为我们是一家非常开放、有朝气、有前途的公司，而且我们的管理非常人性化。”也许正是因为我喜欢探索的性格所致，还不想早早地去大公司过安稳的生活，而且之前已了解到业内对SAP的赞誉声音，再加之得到了Hasso Plattner的重视，综合考虑，终于动心了。虽然那时候还不晓得什么叫ERP，但想想反正才三十岁，还年轻，大不了失败了再加入大公司。现在看来，选择SAP是我这一生中最正确的决定，事实上这很意外，就像是命运的决定。

1992年，我正式加入SAP，担任高级平台软件工程师。由于我懂数据库、网络架构、面向对

象设计等各方面的知识，并能够将这些知识用于开发当时的旗舰产品R3，所以一到公司便得到重用。在那段时间，我一直站在技术的最前线，是坚定的IT实践者，所以我有技术人员的鲜明个性——非常骄傲于软件技术方面的专精，瞧不起管理人员；非常抗拒成为管理者，觉得只有那些学MBA不懂技术的人才去当管理者。

1993年，SAP从总部挑选出六位工程师派到硅谷，我就是其中之一，去建立美国研究院，直接支援美国客户。硅谷是IT圣殿，最新的创新都来自那里，我当时是抱着朝圣的心情去的，希望能够吸收当时最前沿的技术。但到了硅谷后，我才发现还要做很多其他事情。由于R3是世界第一款C/S企业管理软件，其受欢迎程度不逊于现在的iPad，但在美国的服务速度远远跟不上销售速度，也没有专门的技术支持人员，所以我们虽然身为研发人员，仍要从事一些技术支持方面的工作。当时像HP、Apple等公司的管理软件用的都是R3，我还去帮它们建立了整套ERP系统。在那个过程中，我遇到了多方面的挑战，单靠技术只能解决一部分问题，更多的还需要解决技术和产品如何在客户那里得到应用。因此，我的视野慢慢打开，开始用开放的心态，去接受自己观念的转变，对管理的认识不再那么偏激。

华丽转身——从工程师到管理者

1995年，这种转变达到了质变——我被派到中国，后来担任SAP大中国区高级副总裁，负责中国大陆、香港、台湾地区的咨询业务和项目实施。在这段时间，我必须独立面对很多事情，放开心态去解决问题，渐渐从一个完全以技术为导向的人，变成一个越来越走向管理的人。

事实上，我选择回祖国，还有一段蛮有意思的插曲。1994年，Hasso Plattner已经不再做技术管理，而是担任CEO，另外一位董事Peter Zencke负责产品研发，我归他管。那时SAP尚未开拓中国市场，但已有打算在中国落地生根。于是，Peter找到我说：“祥麟，你要不要去？”我说：“我在台湾长大，非常向往祖国大陆，但我从来没去过，我可以去尝试一下。”于是我答应Peter会在三个月内招十个人，把他们训练好，并把公司地点找好，然后就回到硅谷来。结果大家也知道，这三个月一待就是十七年，有时想想也算是一种因缘际会。

我一直很自豪是SAP中国的第一个员工，除了销售之外的其他所有团队，从咨询、培训、客户支持以至产品研发，都是我一手建立起来的。还记得公司办营业执照时，由于国家对独资外企管理很严格，我几乎跑断了腿，去北京各个相关部门盖了一百多个章才拿到执照。

同样，我也很骄傲，埃森哲、IBM等在中国的第一批顾问是我培养起来的。我还开了顾问培训班，并亲自授课。当年的那些顾问现在都发展得不错，在中国咨询界独当一面。所以我很高兴至少在中国，能够桃李满天下，这是我一辈子的骄傲。

1994~2000年，我基本上是从实践中体验真理，从实践中学习管理。到了2000年时，我终于决定要接受一个系统化的管理培训——公司花好几万美元送到我斯坦福大学读一个EMBA课程。想想挺好笑，直到1995年底前，我还瞧不起那些学MBA出身的管理者，而现在我自己却要去读了。其实这也是我心态的转变。公司成长太快，要想成为一名优秀的管理者，单靠从实践中体验真理这条路有一定的局限性，所以需要进行系统的学习，包括财务、人事管理、战略制定、组织结构等各方面的知识。斯坦福带给我很多系统性知识，有些是我过去在摸索中所实践的，只是当时不知道它们已经写在书本上了。

管理哲学——不做代工

虽然到中国后大多时间从事管理方面的工作，但建立SAP中国研究院一直是我的梦想，因为我的根还是在研发。1998年时，SAP计划在中国或印度成立在亚洲的第一个研发中心，这个过程中我一直在全力争取，可惜最后还是落地印度。在此后的五年中，我没有停下争取成立中国研发中心的脚步，2003年，SAP中国研究院终于得以尘埃落定。

当打算成立SAP中国研究院时，我就不希望它是一个代工基地。不管什么规模的软件公司，其最大的资产是人才和知识。招两三千人并不难，但问题是把技术留在中国，把人才培养起来，如果只是按美国或者德国的设计纯粹做编程，一点参与设计的机会也没有，就像在象牙塔里编程，只是达到了量的增加，而没有质的改变。

软件产业从市场需求分析、系统设计、编程、测试、客户支持直到推向市场，是一条完整的价值链。正如Acer施振荣所言，这条价值链就像一条微笑曲线，在开始的创新设计阶段和最后的进入市场阶段，是附加值最高的部分；而中间部分的编程阶段、测试阶段，是附加值比较低的部分。中国软件产业要想脱颖而出，肯定要走曲线的两端。因此，我立定目标，SAP中国研究院必须掌握产品的完整生命周期和核心技术，培养优秀人才，从而维持整个创新的力量。结果也是如此，SAP中国研究院一直在往这方面发展。这也可以称得上是我的管理哲学。

随着近几年管理的深入，我深深体会到，将中国文化与德国文化的精髓相结合应用于管理上能起到事半功倍的效果，因为我所管理的两千多人中95%是中国人，他们一直受到中国文化的熏陶，具备中国文化的基本精神，比如儒家思想、中庸之道等，这些都是优点；而德国文化所具有的实事求是、讲求效率的精神，对于公司的发展也起到了积极的作用。

细数成功路——千里马和伯乐缺一不可

我经常在想，之所以走到今天这一步，取得看起来还算过得去的成绩，一方面是我自己努力和储备的结果，另一方面，也是得到了两位贵人的相助。

第一位是Hasso Plattner。他在我心中是一位领袖级人物，值得我为他抛头颅，洒热血。在我对SAP几乎不感冒的情况下，他慧眼识珠，用其热情深深打动了我，使我加入SAP。在我刚入公司时，Hasso经常和我探讨工作到深夜两三点。对于他想要的结果，我必须马上能够编程实现，然后给他看。如果编写某个程序需要花费一段时间的话，他就蹲在墙角抽烟，等我把东西做出来。他的这种意志力深深感染了我，让我体会到干革命的感觉。因此，我那阵子很疯狂，曾经连续两个礼拜在公司过夜，就睡在公司里面。另外，由于我的个性非常直爽，和他讨论问题的时候常常毫不避讳地指出一些问题，这使得Hasso非常惊讶于我这个中国人有比德国人更直接的个性，也因此对我更刮目相看，在日后委以重任。



2005年，SAP“中国计划”的缔造者曾彼得（Peter Zencke）访问了SAP中国研究院，与我共同探讨如何建立SAP Business ByDesign产品的本地开发团队。

第二位是Peter Zencke。他虽然现在已经退休，但他对技术的把握令我深深佩服。当和他谈论任何一项技术时，他都能够找出问题的关键，给我一种茅塞顿开、醍醐灌顶的感觉。而我在SAP能有今天——回到祖国并有机会建立SAP中国研究院，延续我对技术的追求——也是得到他的信赖和提拔，也许在他心中，第一次遇到我这样一个讲话那么直接的中国人，让他感觉到了德国老乡般的惺惺相惜。

每个人在职场中都希望能够遇到伯乐，但所谓“千里马常有，而伯乐不常有”。我现在已经年长，很希望自己能够成为伯乐，以发掘出更多的千里马，让他们能够独当一面，从而为SAP、为信息化奉献自己的智慧和力量。

英特尔中国研究院院长方之熙

我与研究的不解之缘

口述 / 方之熙　整理 / 董世晓

作为英特尔中国研究院掌门人，方之熙的经历颇为光鲜和丰富，而从他的经历可以看出，“研究”二字注定与他结下了不解之缘。



英特尔中国研究院院长方之熙

从数学到计算机——找到人生的激情

我一直认为数学非常奇妙，各种定理、认证都有一种独特的美。我在高中时就因为参加数学竞赛取得了不错的成绩而得以跳级，后来很幸运地考入复旦大学数学系，接着还考上了许文华老师的数学专业硕士研究生。那时计算机科学开始兴起，复旦大学很多计算机系的教授同时也是数学系的教授，因此我在课堂上也初步接触到了一些计算机的知识，对计算机科学产生了兴趣，对于计算机可能给生产、生活带来的强大影响觉得无比惊奇。

与数学相比，我发现刚刚兴起的计算机科学无论是在研究内容还是研究方法方面都有很大的不同。计算机科学是以应用为目的，它有强大的工业应用为支撑，因此我觉得它应该有不错的前景。

后来，我得到叔叔的资助赴美留学，先是读数学，但发现美国的数学教育并不比国内好，而由于看好计算机的发展前景，于是我就转到了美国中部的内布拉斯加大学林肯分校学习计算机。

在那个年代学习计算机非常辛苦。当时电脑是奢侈品，即使在美国的大学里也只有少量的几台电脑，都需要登记才能使用。记得有一门课需要大量编程序，我几乎经常需要熬通宵写代码，但每当看到一行一行的代码变成一个很好玩的程序时，我再一次体会到了计算机与数学一样的美，因此每次我都非常兴奋。

1984年夏天，本着对计算机的激情，我用不到三年的时间就把博士读完了，成为这所学校里在最短时间内拿到博士学位的中国学生。

从此，我开始了与计算机工业近三十年的共同成长，见证并经历了世界计算机行业持续不断的技术革新和蓬勃发展。

搞研究也要以结果为导向——在美国当教授

我博士毕业那年，全美国只有两百多人获得计算机专业博士学位，所以很好找工作。而我非常喜欢研究，因此毕业后选择了去美国中西部的一所州立大学担任计算机教授。

刚开始教学时，我总是想把原理讲得清清楚楚，但学生却对原理没太大的兴趣。一位年长的美国教授观察了我的授课方式后，问了我一个很有趣的问题。他说：“为什么美国的厨房里有各种各样的刀和锅，而中国的厨房里可能只有一把刀、一只锅？因为美国人习惯学会了各种厨具的用法来制作不同的美食，而中国人则习惯学会用刀与用锅的原理后，只用一把刀、一只锅来烹饪不同美食。”

他的这番话对我的启发很大。美国人的习惯是学会使用工具，强调应用；而中国人讲究摸透原理，强调事情的来龙去脉。计算机是一门应用科学，它并不需要一名本科生从微积分和计算机原理开始学习，而是要全面学会使用工具，重视实际应用，这样出去就能做事，而研究生以上才需要强调基本原理的研究。当时的中国计算机学生普遍理论知识非常扎实，真正在编程能力上却又欠缺。通过这件事，我慢慢体会到了一个道理，那就是即使搞科学研究也得注重“结果导向”，这个理念影响了我后来近三十年的研究生涯。

鼓励冒险成就前进的动力——从创业到英特尔研究院

教了两年多计算机后，我感觉自己离工业界太远，无法将那些理论应用到有价值的实践中去，这让我的研究工作缺少了工业应用这个计算机科学最重要的基础。于是我开始考虑向工业界发展。我先去了伊利诺伊大学香槟分校做博士后研究，因为那里的研究课题非常接近工业应用。

1986年，我做完博士后研究之后，先去了新泽西州一家做火箭发射设施系统的公司工作。但那家公司的高层赚了很多钱，对员工却很苛刻。1988年，它终于到了倒闭的边缘，我便离开这家公司。随后我加入了一家超级计算机公司。时值20世纪80年代末期~90年代初期，随着PC的诞生，超级计算机逐步被取代了，加上美国当时处于金融危机，所以这家超级计算机公司后来举步维艰。我又开始寻找新的机会，去了惠普研究院。

当时的惠普研究院和IBM的麻省研究院很近，都属于美国第二代的研究院。在惠普的经历给我最大的体会就是工业研究院的研究必须结合公司的发展战略，不然做的研究再好，也只是停留在实验室阶段，是没有实际应用价值的。从惠普出来后，我和朋友在硅谷创办了一家公司。但由于对市场把握不好，最后不得不关门大吉。这段经历带给我的最大收获便是要考虑失败——做一件事情，不能只考虑成功，一定要考虑失败了怎么办。如果不这样做的话，失败就比成功的概率大很多，特别是对于小公司而言。

> 回想当年在下放劳动中仍不减对数学与计算机的痴迷，到如今又回到中国管理英特尔全球五大研究院之一的奇妙经历，我相信这真的是一种缘分。

1995年，我正式加入英特尔公司，刚开始是在英特尔惠普系统架构产品部门做系统科学运算方面的工作，主要是将之前在惠普做的构架转变成产品。1996年上半年，英特尔研究院成立，我是第一批加入英特尔研究院的人，当时研究院只有不到二十人。加入研究院后不久，我遇到了在英特尔的第一个大挑战。当年Java横空出世，在软件市场好评如潮。因此，提供性能绝佳的硬件以更好地支持Java运行，



2009年，方之熙与英特尔首席技术官贾斯汀共同宣布英特尔中国研究院成立

成为当时英特尔业务发展的关键。但是，不久后我们就发现Java在英特尔硬件上的运行速度并不理想。由于Java虚拟机以及Just-In-Time Compiler的源程序属于技术机密，因此导致我们难以分析、确定运行不畅的原因到底是在硬件还是软件层面。

针对这个问题，我们当时专门进行了封闭式讨论。我酝酿了很久，向我的领导——现在已是英特尔全球副总裁、首席技术官的贾斯汀提出了一个极具风险的建议：在实验室里自主开发Java虚拟机程序，然后在英特尔的硬件上运行，从而找出运行不畅的问题所在。当时贾斯汀也很犹豫，因为他担心我们能不能够在很短的时间内成功开发出Java。我当时虽然也很担心，但有信心做出一个类似样机的东西。贾斯汀看到我信心满满，当即让我领导这个项目，并给予了各方面的大力支持。

历经9个月的日夜攻关，我们二十多人完成了在外人看来需要调用企业级的力量、组织几百人才能承担的工作，Java虚拟机初步成型。此后又经过近两年的技术攻关，我们终于成功地开发出了Java虚拟机的源程序，找出了问题所在。后来，针对实验中出现的问题，我们分别给客户提供硬件和软件两方面不同的解决方法。这个项目的成功，使我对于英特尔鼓励冒险的企业文化有了更加深刻的认识。风险是一种鞭策力，没有风险，就没有前进的动力。

与中国的不解之缘——创立英特尔首个全球级嵌入式系统研究院

早在1998年，英特尔中国研究院的前身——英特尔中国研究中心就已经在北京成立了，那时候我在美国的研究院负责微处理器实验室，经常与北京的研究员们合作做项目，包括最近刚刚升任我们研究院首位本土首席工程师的吴甘沙。当时感觉在中国本土的这些研究员们非常踏实，有强烈的学习欲望和探索精神。从那时开始，我与中国的研究团队以及学术界的交流和合作就非常频繁了，期间我还促成了英特尔与中国科技部、清华大学合作成立了先进移动计算研究中心。经过这十来年的发展，中国研发实力和创新能力也越来越强，英特尔中国研究中心也在2009年10月正式升级为英特尔中国研究院，成为英特尔全球五大研究院之一。作为第一任院长，我带领近100名优秀研究员面临的第一个难题就是：作为一个全球级别的工业研究院，我们要选择什么研究方向？

经过近半年时间的广泛调研、沟通和讨论，在医疗设备、汽车、云计算等众多提议中

我们选定了中国研究院的研究方向。2010年4月，英特尔CTO贾斯汀和我在北京的英特尔信息技术峰会（IDF）上共同宣布了英特尔中国研究院的研究战略——致力于打造世界一流的嵌入式系统研究院。

为什么要专注于嵌入式系统研究?

首先，近年来中国在嵌入式领域内成长快速。中国的手机、互联网、有线电视用户人数均居全球第一，也已成为名副其实的世界第一汽车市场。从智能交通到车载信息设备，从“物联网”到智能家电，嵌入式移动互联在中国发展前景广阔，这为我们提供了很多非常好的研究课题。并且这些研究课题在世界上也都是非常前沿的，针对这些课题的研究将直接对中国的市场与经济发展产生积极的影响，这也是我们“立足中国、贡献中国”的宗旨所在。

另外，这也符合英特尔的全球战略。我们当时成立了顾问委员会，请了几位工业界、学术界最有成就的国内外专家帮我们做审核，一起看这个方向是否正确。最终大家发现，在汽车上也好，在消费电子、医疗器械上也好，都是属于嵌入式系统的问题。我们把这些整合在一起，认识到面向嵌入式系统的研究是英特尔转型中的重要任务，也是我们要做的事情。

最后，我们在人才建设方面也已经有了一定的积累。我们的研究员在编程技术、多媒体技术、硬件设计、无线通信技术等方面具有深厚的技术功底。而这些技术专长正是嵌入式研究的关键基础。

现在我们已经建立起了一支完整的嵌入式系统研究团队，全面涵盖嵌入式应用、互连嵌入式技术、嵌入式软件、嵌入式架构、嵌入式输入输出技术、先进移动技术等领域，致力于为全球及中国市场提供突破性的手持、消费电子及嵌入式技术。目前我们已经开展了数十项行业领先的研究项目，其中“个性化电视广告节目推荐”等近十项研究成果已经投入产品开发阶段，“英特尔架构上的无线基础设施”等技术已投入实际应用。

从事了三十多年的技术研究，每当看到人们的工作与生活因为我们的创新技术而提高时，我都感到一种无法言表的成就感。因此，我特别重视与研究员一起分享研究的价值与快乐。因为在我刚担任院长时，发现很多新入职的研究员在职业规划上都希望做高层管理人员，但对研究本身却缺乏足够的热情，这其实非常不利于他们的成长。因为搞科研需要耐得住寂寞，要有强大的自我驱动力。为此，我们根据每位研究员的不同特点，为他们安排了合适的导师。导师一般都是资深的研究人员，是研究员眼中的偶像。通过与偶像交流技术、畅谈人生，我们的中国研究员也逐渐把技术创新作为自己的人生追求，并开始体会当自己的技术得到运用时所带来的巨大成就感。

现在，英特尔中国研究院正在健康、快速地成长，这让我非常欣慰。回想当年在下放劳动中仍不减对数学与计算机的痴迷，到如今又回到中国管理英特尔全球五大研究院之一的奇妙经历，我相信这真的是一种缘分。它为我一生带来了源源不断的激情，引领我勇敢并平静地接受每一次挑战，珍惜每一个机遇。



方之熙在IDF2011技术研究前瞻日上做演讲

微软大中华区CTO 张湘辉

技术成长的成功之路

整理 / 高松

张湘辉，1994年加盟微软，现任微软大中华区CTO。本次采访中，张湘晖介绍了他在技术方面的成长经历，管理经验，以及心得感悟。

走近张湘辉

程序员（以下简称“程”）：您最早怎么接触到计算机的？

张湘辉（以下简称“张”）：接触计算机挺偶然的，完全因为生活所迫，觉得计算机行业的工作环境比较舒服，都是待在大机房，穿白大褂，有空调，所以向往这种好的工作环境，而并不知道电脑是什么，要选什么专业。现在看来还是选对了。

程：早年的经历，对您影响最大的事和人是什么？

张：影响最大的事情与电脑无关。可能和老师有关，我从小没好好读书，当时突然来了一位英语老师，他的英语讲得特别好。后来才了解到，他从外交部下放到湖南，就留下来做了英语老师。这件事对我的触动很大，第一次觉得别的语言可以那么美。

程：当时怎么选择技术方向的呢？

张：大学对学习软件兴趣比较大，后来到美国，就毫不犹豫地选了软件开发。这个过程中，接触到一位老师的研究还挺有意思的：他考虑如何把软件开发的整个生命周期，尽量做得有一致性，即你要做什么，结果出来就是什么，这里面应用了大量人工智能的技术，所以就师从于这位老师，读了软件工程的博士。

程：这期间您有什么收获呢？

张：读博士时对软件产业和软件开发过程中的问题有了较多了解。读完博士之后，尽管有些大学给了我教职，但也没有去大学教书，因为还是想做些实业，于是进了一家在西雅图的软件公司Aldus，它之后被Adobe收购，我就到Adobe做软件开发与设计。当时有一款制图软件（类似于Photoshop），叫PhotoStyler，这是由一家名为Ulead台湾公司研发的（最后被Corel收购）。当年他们的创新能力是很强的，所以我跟他们配合一起来做PhotoStyler，这真是受益匪浅。

首先让我了解到软件开发的整个过程，原来读书时学的都是纸上谈兵，那时开始遇到了实际问题，并去解决。这段时间里，更多的是看到中国人在做软件方面的才智。Ulead实力非常强，当时在中国大陆做软件的还没有那么多，但从他们那里可以看出我们中国人做软件是很有潜力的。

跟他们共事一段时间后，我想一定要回国做软件。所以后来听说微软在招人建立研发中心，我就很踊跃地申请了，这对我来说简直是梦想成真。到微软工作，同时又在中国，这是一件非常完美的事情。

回国做软件

程：来中国做软件，您对微软这个决定如何理解呢？

张：我在1994年8月底被派回来创建研发中心，这充分体现出了微软对中国市场的觉醒。微软当时在做Windows 3.2，这是一款划时代的产品，也是我回国后的第一款产品。我带领

了一支非常年轻的队伍，在美国版发行六个月以后，发布了中文版。中国用户第一次如此之快，就享受到中文版的微软产品，此后中文版与英文版的发行时间差越来越小，现在几乎是和英文版同步发行了。

上面提到的时间差跟软件开发实践是有关系的。如果一心做美国版，不涉及其他语种，等做中文版时，就会发现有很多东西是原来设计所没有考虑的，比如对一些字体的支持，对一些输入法的支持等种种问题。特别是对汉字的支持，早期的GB-2313字符集共包含6763个汉字，这导致最早的软件里面会有一些中文显示不出来。所以，要从上游来解决所谓时间差太长的问题，从最早期开始考虑对中文版的支持。这就要求中国团队能够和美国版的团队有机配合，现在这个配合是越做越好了。

因为前面的事情做得比较好，那么一些重大的问题，其实在做第一个版时就已经解决了，当然这与技术标准的进步也有很大的关系，比如Unicode的出现，早期支持两万多汉字，到今天支持十万多个汉字，今天谁要编撰《康熙字典》，用我们这套系统，也不需要再加字。从这点看，对弘扬中国文化是很有益处的。

程：刚来到微软中国还是遇到很多困难吧？

张：当时确实很困难，但也挺有意思。最早的研发中心设在科学院第二招待所，那里原来是个歌舞厅，上头还有灯在转。别的不说，就是供电也是很大的问题，当时电的质量很差，所以必须要有稳压器，它能发出很可怕的呜呜的声音，而稳压器本身也过热，又专门买电扇吹稳压器，这哪里像开发世界一流软件的地方，当时心里还非常犯嘀咕。虽然条件差，但我们做软件采取的方法却是最先进的。此外，我们专门租了一条直连美国的网线，那么快的网速在当时中国绝无仅有，以此保证开发的及时和高质量。

程：作为见证者，你也看到了微软在中国的发展，请回顾一下期间微软的技术变迁是怎样的？

张：从研发的角度说，刚开始我们在中国做测试和汉化。之后开始做很多中文项目，并组建起庞大的科研队伍，研究怎样能够将中文处理更好、查找更方便、输入更方便等方面，并开始有些中文版独特的功能。1999年3月，微软宣布维纳斯计划，我们将比尔·盖茨请到深圳参加发布会，这具有象征意义的举动，还只是第一步，能够看出为中国市场做产品，这样的努力一直都在持续着。


张湘辉表示，中国市场做产品，这样的努力一直都在持续着

我印象最深的，是在1998～1999年之前，微软对Internet 还不是很重视，用有些人的话讲就是“不赚钱”。但是微软这样一家几万人的公司，当时做出了一个决策：Everything is Internet，这是殊为不易的，这件事情给我很大的启示。从此，在Internet领域，微软从不是玩家到第一玩家，从操作系统、浏览器到以后不断涌现的新产品，我的体会是微软在拥抱新技术时，虽然及时性没那么好，但是一步一个脚印，踏踏实实去做，最终还是很有信心能赢过来的。

随着移动计算、智能手机，以及平板电

脑的流行，特别是云计算这样颠覆性模式的出现，IT完成了从量变到质变的过程，这是大家有目共睹的，微软在这中间则一直是很重要的玩家。

程：您是如何这个管理庞大的研发机构，并形成有效的技术沉淀的？

张：这是有关软件开发周期的管理问题。做软件与做工程设计有很大的差别，因为其中人为因素很多。有人不再把软件定义为一种科学或工程，而是一门艺术，经常一两个人可以做出很漂亮的软件，而这样设计飞机几乎是不可能的。设计飞机有一套工程方式，可以多加人手，加快效率，做软件则相当不同。当然不是说加人没有用，做大的软件项目，直觉往往是多加点人，但这跟谱好曲子一样，十个作曲家也没法取代一位贝多芬，不过你可以通过很好的管理，让他们共同完成一支很好的曲子，甚至媲美贝多芬。这就是软件管理的精髓，微软在这方面有很多的创新。

以Windows 7为例，包含七八千万条甚至上亿条代码，五六千人同时开发，还有很多合作伙伴确保周边产品兼容。对这样一个超大的项目而言，不能一眼盯到结果，不能像跑百米一样，始终盯着终点。我们的经验是盯终点肯定乱，因为要经历非常漫长的过程。

从心理上说，当发现离终点还很遥远时，人就会泄气，不能以那么快的速度玩命跑下去。最好的方式，是将事情分成很多步骤来做。Windows7从开始到完成可能要耗时两年，以两年时间为一个周期，那么前六个月团队就会被弄垮，所以你必须以也许每两个月为一个终点。就像跑一千五百米，我们要考虑第一圈跑多快，第二圈跑多快。

这就需要把每个终点区分得很好，设定有效的里程碑，在逻辑上要很精准，是不是到了这个里程碑，同样要非常清楚。这样每个里程碑达到时，大家可以庆祝一下，重又奔向下个目标。如同爬珠穆朗玛峰，没有说不断爬上去，而是先到大本营，再到第几个营地，最后才能登顶。

设定里程碑是横向的时间划分，纵向来讲是人的管理。比如，怎样找到问题，找到问题又交给谁去解决，怎样互相有效交流，怎样互相检查代码，这方面的很多做法，我觉得微软到今天仍是独一无二的。

同时，今天的软件开发实践跟原来有些不同，早期网络没这么发达，必须把一件事完成才能发布。在Internet永远是Beta，也就是把产品阶段分得更细，这是一个精细而完善的发布过程。用户也更喜欢高密度的发布。但是要注意重大功能出现的时候，还是要从总体设计，而不是在结构没有很大变化的情况下，突然塞进一个很大的新功能，这需要考虑到软件模块设计等因素。

关注与推荐

程：您目前关注哪些技术？

张：云计算及其生态系统，是非常值得关注的。IaaS能够把现有的计算中心、数据中心的效率提高，将管理水平提升上去。要提供更多适合于云的应用服务，建立这样的生态系统需要时间过渡。

我们认为在云时代的整个生态系统里，通吃是不可能的，没有一家公司能够做到，要提供最好的用户体验，就要求大家用通用的标准去开平台环境，在其上开发应用，厂商之间还要有很好的默契。通过和客户打交道，很多人都要求支持混源（MixSource）的环境（有的用Windows，有的用Unix/Linux），在不同环境下开出来的应用，还能够很好地有机结合在一起，这样的实际需求是很大的。对此我们做了大量的工作，数据级、程序级的互操作都做得很好。所以在Azure平台上，无论是使用 Java、PHP、Ruby，还是.NET Famework等写的东西，都能得到应用。

另外，提高用户体验也是热点，比如支持多点触屏的智能手机，能提供更加贴近用户习惯的体验。再比如，人机交互过程已经从鼠标过渡到语音、肢体，甚至表情，微软的Kinect就是非常好的例证，仅靠动作就可以完成人机交互。这方面的进步是突飞猛进的，也将给技术人员更大的空间，做出更有趣的东西。

神州数码信息服务集团副总裁谢耘

敢于挑战所有未知领域

记者 / 陈秋歌

敢想、敢干，经受得住各种挑战的技术管理者，才能称得上优秀；对于“跨行业”的管理者来说，获取员工的信任，是成功管理的关键。

神州数码信息服务集团副总裁、神州数码工程院院长谢耘，毕业于清华大学，并获得博士学位。他主持研发了国内第一套有线电视加密收费系统、国内第一台基于Windows CE的掌上电脑等，并著有《修炼——我的职场十年》和《成长》两部以职场感悟为主的畅销书籍。一次偶然的机会，他走向了管理道路，从此踏上了多次“跨行业”管理，甚至是“外行人”管内行人的管理之路。相信他的管理风格和管理经历，可以给其他管理者一些启示。为此《程序员》杂志专访了谢耘博士，让他为我们分享一下他不寻常的技术管理精英之路。



谢耘表示为了项目做好，唯一能做的就是“信任”，敢于把自己的命运交给下属

记者：您的中学时代是如何度过的？您最擅长哪些科目？

谢耘：回忆起中学时代，我觉得过得还是比较轻松的，“一点苦也没吃过”。我小时候身体不太好，所以父母在学习上没给我任何压力。也许正是这种宽松的家庭教育环境，让我对学习没有任何负担，可以做自己感兴趣的事。

我父母都是大学老师，从事机械教育工作。受他们的影响，我从小对科学很感兴趣，比较喜欢能感知到的事物，喜欢分析问题，解决真实世界中的问题。因此我的物理学得非常好。一般的题目只要扫一遍，我就能在脑海中把它解答出来，几乎不用笔算。对比较抽象的数学科目，除了应用部分外，我就不是非常喜欢学习了。

中学学习政治时，我从来不背政治，都靠分析。记得高考前，我母亲因为我不背政治，很着急，甚至找到了我的政治老师。我政治老师告诉她：“你儿子不用背，他会分析。” 因为学明白了政治，所以后来我的哲学学得也非常好。这对我影响还是很大的。所以我一直提倡学生时代还是要好好学习马克思主义哲学。

在学校，由于学习成绩好，从小老师就鼓励我去帮助那些学习上有困难的同学。所以我被“练就”了一副“伶俐口齿”，知道如何把一个问题阐述清楚，以便对方明白。

后来，我以全校第一名顺利考入清华大学。记得高考刚结束后，同学们都说我考得最好，因为监考老师说考场中就我最镇静。有许多人一有

压力就紧张。其实在压力大到一定程度前，我也紧张，但超过某一界限时，我反而不紧张了。这也许是天性吧！这种天性对我后来跨越一个个职业难坎也有很大的影响。

记者：请您谈谈在清华大学的求学经历。

谢耘：告别了轻松的中学时代，我进入了清华大学电子系。可以说当时清华大学是国内同类学校中最累的一所，而电子系又属学校中最累的几个系之一。

清华大学本科课程非常重，老师给的压力也很大，我不得不用功学习各个科目。记得一次期中物理考试结果出来后，由于我们考试分数特别高，因此物理老师对所考试题不太满意。期末考试，我们出现了一堆不及格。事后发现老师竟然直接拿了物理系的考卷来考我们。一次数学分析课考试，全系三百二十多个人中只有五个人答对了最后一道题。我算其中的一个。20年后，我去看这位数学老师，他对这件事仍记忆犹新，还说那是故意考验我们。

终于熬过了清华大学繁重的课程压力，五年的课程，我学了四年半便提前毕业了。由于功课比较好等原因，我继续留在清华大学攻读硕士和博士。读博期间，我在学校兼任政治辅导员，负责学生的日常管理工作。这段经历是我走向管理的第一步。

读博士期间，我被公派到了悉尼大学。除了接触到当时还比较简单的互联网（像电子邮件等）外，我并没感觉到国外大学与国内有什么太大区别。在我看来，当时的悉尼大学和清华大学水平相当。既然拿着国家的钱出国学习，就要对国家讲信誉。正如我向导师承诺的那样，在悉尼呆了两年，做完课题后，我便选择了回国。

拿到清华大学的博士学位后，由于多方面原因，我没有接受学校的挽留去了一家也符合公派留学相关要求的国有企业。至此便开始了我人生中的职业打拼生涯。

记者：清华大学博士毕业后，您从事的第一份工作是什么？当时担任什么职位，主要从事哪方面的技术研发工作？

谢耘：博士毕业后，我去了一家北京无线电厂，在那从事研制雷达工作。限于当时国内条件，新研发项目审批迟迟下不来，在那呆了一年后，我离开了。

一个同学所在的公司要与美国一家大型国防企业TRW合作开发一套有线电视加密收费系统。鉴于我有几年的留学经历，又是技术出身，对方邀请我去负责技术管理。可以说正是这次偶然的机会，让我第一次踏上了管理者之路。

1993年底，我带领当时的项目团队去了洛杉矶，开始与这家美国公司协作开发此项目。TRW属于美国当时的顶尖企业之一，拥有美国一流的技术高手。该公司当时负责这个项目的带头人是一个台湾博士，姓周。他聪明敢干，是我见过的为数不多的几个天才之一。周博士之所以能在TRW脱颖而出，据说是因为他曾做成了一件别人都认为不可能的事情。事后回想，我非常庆幸自己刚刚走上社会不久，便能和周博士这样杰出的人才共同工作。周博士用自己的言语、行动和亲身经历让我意识到自信和勇于挑战对成功的重要意义。**如果你认为自己优秀，就应该足够自信，要相信自己能完成他人做不到的事情**。这一信念使得我在后来面对各种别人认为不可能完成的任务的时候，从来没有退缩过。而且完成这些不可能的任务，成为我工作的主要内容。

一个人在年轻的时候，在还没有被自己的经验固化的时候，能够与一流的高手共事过招，我认为意义极为重大。这个系统在研发的过程中，出现了一个非常致命的问题----电视屏幕中间出现了一道黑。当时核心芯片是TRW做的，但他们并不知道问题出在哪里。和我同往的一个同事看出了问题所在。虽然我不懂芯片，但是根据常识和逻辑分析，我认为这个同事说的没错。由于我英语比较好，所以由我来代表中方项目组向TRW公司的工程师解释。当时我们从晚上十点一直开会到第二天凌晨三点，最后除了一个美国工程师外，其他人都表示认同中方的想法。第二天，按我们的想法去做，问题解决了！这件事让我意识到美国技术高手解决不了的问题，我们照样可以解决。技术道路上没有什么可怕的，我们大可不必胆怯、自卑。

因为我是学电子工程出身，并不懂电视。这段经历也是我第一次作为外行人去管理一个团

队。为了把这个项目做好，我首先做的就是“信任”，知人善用，并且敢于把自己的命运交给下属。在实际工作中，我对属下非常放权，该他们管的事放权让他们去做。他们处理不了，需要协调支持时，我才会出面。我认为这一点对于一个外行管理者很重要，否则你的管理将无法进行。当然在这个过程中，你也必须努力学习，让自己尽快成为一个内行人。

记者：您什么时候进入联想的？在联想，您曾组织开发了中国第一台基于Windows CE的掌上电脑。请您介绍一下当时的开发过程。

谢耘：几经周折，甚至“失业”一段时间后，1998年1月，我第二次进入联想，负责做掌上电脑。掌上电脑对于我来说，是一个新事物。对这行，我完全没有经验。这样我再一次投入到一个对我来讲完全陌生的领域，开始了又一次新的跨行业转变。

掌上电脑该如何做？当时这个产品在国外也处于开发阶段，尚未投放市场。我们除了网上一张照片可供参考产品大小之外，没有其他任何资料实物能够借鉴。当时的产品需要使用微软Windows CE操作系统的汉化版本，可当时该操作系统还没有中文版。于是我们几乎是与微软公司同步对Windows CE进行了汉化。

产品研发工作也曾因讨论选用哪款CPU一度陷入僵局，最后我在听取同事对当时三种CPU性能评估结果的基础上，决定采用飞利浦的嵌入式应用CPU。对于这个决定，我也没有十足的把握，只是按照同事给我的分析报告确定了一个相对好的方案而已。至此，产品的整体方案基本确定了下来，开始进入实质性的设计工作。

克服了技术上的种种困难，1998年7月2日，第一台基于WindowsCE的掌上电脑样机终于面世。这个产品引发了国内及国际媒体的极大关注。也因此，我们这十几个人的团队引起了公司最高层的注意。这件事让我意识到，**利用阶段性结果，去推动影响工作的因素向积极的方面转化，是管理中的一个重要技巧**。

1998年10月我们开始了该产品的试生产，并于1999年投入市场。虽然产品并不是非常完善，但仍供不应求。

记者：神州数码从联想集团分拆出来后，您便担任副总裁兼首席技术官，这时您的工作重点是什么？

谢耘：2000年，联想分拆之后，我留在了神州数码担任CTO，开始了技术管理生涯。。

记得2002年我接手软件公司时，公司员工技能普遍老化，500名技术人员中竟没有一人懂Java。同时，公司也缺乏具备核心竞争力的优势产品。能否把这家企业救活，我的下属员工曾对我这个从没从事过行业软件及解决方案业务管理工作的“外行人”产生了质疑。在仔细分析了公司的核心问题之后，我从提高员工技能入手，专门请了一家新加坡公司来给员工培训。为打造公司核心竞争力，我们在当年立项了五个研发项目。当时这些项目都没有现成的客户需求，只是从市场分析的角度觉得可行。令人可喜的是，当年这五个项目就卖出了四个。其中一个项目卖出时还仅仅只是一个PPT方案：CTAIS2.0——国税征管系统。后来这个系统被国税总局采用，在全国范围内进行了推广应用。就这样靠研发救活了软件公司。公司运营一切正常后，2004年——我工作十周年之际，我利用闲暇时间写了《修炼－－我的职场十年》一书，主要回顾和总结了我从业十年来的人生经历及诸多感悟。

2007年，我应公司业务发展需求担任神州数码工程院院长，主抓研发管理工作，直到今天。

结语

从当初一个偶然的机会，谢耘从技术转向做管理，并在管理道路上越走越远，期间曾接手管理了多家不同行业的公司，在管理方面积累了不少经验。在他看来，做好技术管理工作，首先要心胸开阔、开放，尤其对于一个外行管理者；其次要从哲学的、辩证的角度来思考问题，不能把一个问题绝对化；其次就是要能够接受很多“不公平”和“委屈”，一切要从公司大局出发；最后，要有各方面的、综合的知识背景，这可以帮你快速理解所遇到的事情。所以建议大家不要盲目从事管理工作，首先衡量一下自己是不是适合、喜欢做这项工作。

Autodesk中国研究院核心产品及运营资深总监云浪生

成功是一种习惯

记者 / 董世晓

作为Autodesk中国研究院核心产品及运营资深总监，云浪生具有丰富的技术团队管理经验。近日，他与我们分享了在技术管理方面的成功经验。

记者：从童年到现在，有哪些重要的人和事让您印象深刻？

云浪生：我记忆中好像没有什么重大事件，倒是生活中长期耳濡目染的小事对我影响很大。

从小受父母和家庭的影响，我对老师和具有专业知识的人都很尊敬，从而深深地影响了我对技术专业的热爱。

让我留下深刻影响的是那些总有着忘我精神去最求的人。

记得有一位同事离职创业之前，大家根本感觉不到他有任何变化，工作还是一如既往的认真，还是那么有激情。

记者：在从技术工程师到技术管理者的转变过程中，应当锻炼自己哪些方面的能力？

云浪生：对于这种角色的转变，每个人都有自己不同的体会，而我更注重对性格的铸造，因为它比技能的提高更难，但同时也更有效。或许用磨炼这个词可能更贴切一些。

学会宽容。允许和鼓励在技术上较真，而在人的问题上，则不能太较真儿，除非涉及原则性的问题。

坚持不懈。像南非前总统Mandela那样，要有点能把牢底坐穿的精神。我们往往可以对一个技术问题锲而不舍，但对生活问题和工作问题比较难做到。

积极、正面地看待问题。记住，挑战永远意味着机会。

热情。毛主席说过：“人是要有点精神的！”有热情，才能去追求卓越，才能锲而不舍，才有可能感染他人。

有广泛的阅读和学习的兴趣。这点也很重要，因为它是厚积薄发的基础。

最后一点，慢慢地把上述的态度和行为都养成一种习惯——**成功就是一种习惯**。

记者：在Autodesk，您是如何做好团队的技术管理，形成有效的技术沉淀？又是如何建立优秀的研发团队文化，使之更有效地工作？

云浪生：Autodesk作为一家老牌的软件公司，有很强的文化传统。比如说，宽松的工作环境，给个人充足的发挥空间，互相尊重，强调团队协作，注重自我管理，不拘一格的民主作风等。这些文化传统是很容易被我们年轻的团队所接受和喜欢，也是我们应该去提倡的。

而要使企业文化的作用发挥到极致，关键在于我们是否能把这些东西和团队人员的具体情况结合好，比如说，我们有宽松的工作环境，但是我们需要有足够的敬业精神并有效地在这种环境中发挥；在提倡创新和变化的同时，也需要团队在技术领域进行更多的积累，在业务领域有更多的了解，并需要我们以更强的主人翁姿态来推动和面对各种变化。

要做到这些，就要从日常工作做起，去做一些经常性的活动。比如说，我们团队设立了一些奖项，为了倡导大家参加团队的管理，我们将奖项的提名、评比都透明化，由团队成员来参与决定。坚持长年做下去，就会有形成文化的功效。



Autodesk中国研究院核心产品及运营资深总监云浪生：成功是一种习惯

记者：每一位技术管理者都有不同的风格，或喜欢超脱、遥远的未来，或基于现实做好本职的工作。请问您感觉自己属于哪种风格?

云浪生：我想我的风格可能是更现实一些吧。做技术软件开发是一件非常严谨的工作，但同时也需要想象力和热情去激励自己和他人，去超越日常中的自我。

记者：您现在关注哪些方面的技术？这些技术要想得到普及，需要经历怎样的过程?

云浪生：我是做设计软件的，我关心的大体都和这有关联。一方面是设计技术的创新，比如说现实捕捉、虚拟现实、增强现实对设计的影响；另一方面，计算技术的创新对设计的影响，比如说云计算、无线计算会怎样去改变设计技术的行为。后者的影响会更为深刻，因为它会改变我们的习惯和商业模式。现在云计算、无线计算，真正的工业应用很少，有很大的发展空间。但要做好，也不是件容易的事。它需要发展整个应用的生态系统。

技术的成熟过程会经几个阶段。新技术出现时，人们会有很高的期望热情。但刚开始，因它不成熟，试了一段时间后会引起很多失望。然后再经过一段时间的改进后，它才会被真正的普及——如果是一个真正有用的技术的话。很多公司并购一些技术也会经过这样的一个过程。

搜狗CEO王小川

不做规划 也要步步为营

记者 / 佳琦

王小川曾负责过孙悟空搜索引擎、内容发布系统、校友录、搜索系统的开发。技术管理者如何完善企业的基因，塑造企业的气质，本篇采访或者会给读者一些启示。

“我现在需要做的是把搜狗的基因完整起来”，刚开始接受采访的王小川如此定调。那什么是搜狗的基因呢?

年少得志

王小川，无疑是少年得志的代表，高中时候，获国际信息学奥林匹克竞赛金牌，在清华大学读自己爱好的计算机专业，还兼任奥赛集训队的教练，此期间以兼职身份加入初创的ChinaRen公司。自言爱好“搞计算机这档子事儿”的王小川归结自己的爱好、学业和工作都在一个方向，感觉比较幸福。

这些得益于他很小就开始编程，小学二年级的他第一次看到计算机就觉得好玩儿，于是从书法兴趣小组转向计算机，当时学习的是LOGO语言，短短几年，就能赢过中学生。上中学以后他更是一直获得成都市计算机编程第一名。王小川当时使用相当简单的设备和系统编程，比如初中时给学校的卡拉OK大奖做积分程序、在计算机上做出了整套的数学问题的机器证明。但王小川从没觉得自己是个纯粹写程序的技术人，只是在不断实践中得乐其中，除了“好玩儿”，王小川说还很享受用自己的智慧，使机器变得更聪明、更智慧的整个过程。

步步为营

自称从来不做职业规划的王小川，还在大四的时候就加入了ChinaRen，原因嘛，据他讲无非是离学校近，并且很多同学也都去了那里。但实际的困难远比提到的便利要多，整个团队都在一个小小的办公室里，当时的王小川负责内容发布系统，因为各方面条件很差，比如需要拨号上网，以致于他每次把编辑文章发到网上去，都得让其他人暂停上网。这就样投入到紧张的工作中，每天睡得不足五小时，以非常高涨的热情，坚持了半年。最终功夫不负空心人，ChinaRen校友录快速成长，取得非常大成功，拥有了一两百万级的用户。

2000年，ChinaRen被搜狐收购，王小川也随着到了搜狐，仍负责ChinaRen的工作。也正是这段时间第一轮互联网泡沫兴起，王小川见证了太多人的放弃、离开公司。“有时候想得太多，反而失去自身的价值“，王小川说，并认为在公司取得阶段性成功之前要耐得住寂寞。

2003年王小川大学毕业，在做了三年兼职工作之后，他正式加入了搜狐。而此前他已经是ChinaRen团队里技术主管了，作为技术出身的人，王小川在此时完成了由单纯的技术人向管理人的转变，他坦言，当时看到手下人写程序花费更长的时间，错误率也比自己高时会很痛苦，但是他也明白几个人加起来绝对比一个人更有效。他开始逐渐放弃自己最擅长的编码工作，而去协调和管理技术工作。但这样的工作，王小川一开始干得并不顺利，“技术味儿太重，让编辑学HTML，这就是我那时干出来的事儿。”但王小川表示现在一定不会这么做了。

当时的王小川仍然负责内容发布系统，并做出了国内第一个可以留言评论的新闻系统，

国内第一个嵌套式的专题系统，同时也在开发推荐系统，这些基于语义关联的推荐，为后来他做搜索打下了基础。然而，王小川认为无论是做内容发布系统还是做搜索，都是搜狐的规定动作，必须要做，而且要做好，尤其在搜索这种技术含量奇高的领域。但当时搜狐能给他提供的条件十分有限。为了做搜索，王小川带着“六个人头”的招人名额，从搜狐搬出来，在清华边上租了一个办公室，从购置办公用品、到给大家提供生活上的便利，一 一操办，还从清华的老集训队里拉了一批兼职过来，一切从头做起，十一个月之后他们的搜索引擎发布了。

虽然当时的搜索没有做到很大的市场份额，但是为王小川积累下了非常优秀的产品团队，他们既有做搜索，也有做客户端的能力，为后来王小川做输入法提供了非常可靠的基础，并且使得公司得以从技术导向转向产品导向。于是王小川开始更多地去分析和学习用户的需求，他的“技术味儿”开始不那么强了。

自己组建团队，做搜索，这让王小川更多地学会了如何与技术尖子们共事。他说，“要把一个人的创造性发挥出来，就不能对他过多要求”，这种放权式的管理，使得团队里的每一个人都觉得处在一个可以发挥自己能力的环境中，而不是在完成领导要求的课题。不但在管理上充分考虑团队成员的需求，团队招人的过程中，王小川也有自己独特的风格：首先要愿意干活儿，觉得是来到一个自己愿意发挥能力的地方，而不是只为寻找一份工作；其次要有奉献精神，坚持诚信；最后能够懂得工作怎样才能做好，并用各种方法去完成。

布局发展

2006年搜狗输入法发布，谈到为何要做输入法，王小川说，在搜狗内部一直强调从技术上思考一件事，看自己有什么优势来做这件事儿。在做搜索的时候经过对网页的分析，以及对大量词频的分析，让搜狗掌握了应用搜索技术为输入法提供准确海量词库的技术。这一次，王小川又让机器变得更聪明了一些。



王小川说还很享受用自己的智慧，使机器变得更聪明的整个过程

2008年搜狗浏览器推出，王小川认为，对于中国人来说完成搜索需要三个产品：搜索引擎、输入法、和浏览器。这三个都是进入互联网的入口，前两个解决的是输入信息和寻找信息的问题，而浏览器则成为连接两者之间的桥梁，这三者的开发都可以用为了“更快的搜索”来解释，让用户更快地去享受互联网。

每一家企业都有自己的优势和核心能力，但如果互相间没有沟通，甚至变成互设禁区不兼容，结果用户需求满足不了，最终大家都受伤。对此王小川直言，目前国内要做我搭台你唱戏的开放平台还为时尚早：首先需要解决的是信任问题，如此才能有效合作，而这将是一场破冰的过程。作为工具，搜狗所有的产品都是帮助用户去访问和享受其他网站服务的，这决定了搜狗的发展模式，也回答了我们开头的疑问，搜狗的基因是技术导向，技术创新和开放，王小川如是说。

豆瓣网首席架构师洪强宁

寻找技术味儿相投的人

文 / 高松

洪强宁及其技术团队在网站架构、性能、可伸缩性上有着深入研究。他们始终致力于用技术改善人们的文化和生活品质，用自己的技术口味去传播分享更好的技术。



洪强宁坦言，如何把以往的技术积累下来转变成产品，是目前摆在自己面前最核心的问题

加入豆瓣

2005年初，阿北在啄木鸟社区发出豆瓣网上线的通知，那时洪强宁加入啄木鸟社区，并学习使用Python已经有三年时间了。

与同龄人相比，因为受学计算机的哥哥的影响，洪强宁上小学时就开始接触计算机，并且很早就有写程序的爱好，并确定了做软件研发工作的方向。在清华的开放实验室里（全国最早允许学生访问互联网的地方），洪强宁接触到了大量国外的网站和技术。那是“我技术能力成长最快的时间”，洪强宁说。大四时候，洪强宁开始了解Linux，学习和使用了后来在豆瓣广泛使用的Gentoo，由此对开源软件产生了浓厚的兴趣，并参与到一些开源项目中去。

清华毕业后，洪强宁一直做嵌入式系统。出于对软件开发的喜爱，他在2002年开始接触Python语言，但只是业余时间才用Python做一些东西。进入第二家公司后，洪强宁在负责新的项目技术选型时，完全采用了Python语言，并应用在嵌入式XP系统上。就这样他逐步完成了硬件工程师到软件工程师的转变，也让洪强宁对一种语言在计算机底层如何工作有了深入的理解，从此无论是工作还是业余项目都使用Python。

2004年8月啄木鸟社区开始运营，这是由黄冬发起的Python语言社区，同在社区里活跃的洪强宁，最终被寻找技术伙伴的阿北发现，经过几次沟通，洪强宁成为豆瓣第一位全职员工。

洪强宁评论这次经验时说，选择一种技术，其实也在被技术选择，进入到和自己有相同技术味儿的圈子。他评价之所以会和阿北共事，也是因为两人在技术口味上有相似的地方，比如同样热爱开源，喜欢Python和Gentoo等等。

除了Python以外，工作中还需要应用的其他技术，如数据库、缓存、分布式系统等等，这些都是洪强宁之前很少接触的，全部在工作中一边做一边学起来。

因为参与社区而加入豆瓣的洪强宁，也与社区一同成长，虽然作为话题参与者的机会减少了，但作为社区建设者的比重却在增加。进入豆瓣的员工很多也都是从Python社区而来，豆瓣也支持员工参与社区讨论，并组织了多次啄木鸟社区的线下活动，而且接管了社区服务器，以至在中国提到Python，几乎到了不能不提豆瓣的地步。可以说，豆瓣成为国内互联网

使用Python语言的一面旗帜。

到豆瓣做改变

洪强宁来到豆瓣前，阿北已经形成了目前豆瓣整套的服务架构，并考虑了未来规模扩大后的影响。

那么洪强宁来到豆瓣之后将如何做呢?

除了负责系统平台、提供技术设施和保证系统运维外，洪强宁及其领导的团队，将阿北原有的架构体系中的所有部分都改造成分布式的，包括架构、数据库、缓存和应用服务器，并开发工具软件去简化管理和维护的问题。原有的技术选型，也不总是能满足不断增长的性能要求，MySQL对服务的支持存在问题，于是他们自己开发了Beans DB。最早使用的是单机服务器，还没有消息队列的概念，这也是后来才加入的。

就这样，豆瓣的改变一步步得以实现，并和技术创新一起持续着。

2007年11月13日，豆瓣用户达到一百万。洪强宁表示，那时压力的确很大，做技术的只有两三个人，大家总是跟在事件之后跑，还没有时间去考虑技术储备，常常是发生什么问题，就赶紧去查找原因，然后设计解决方案，熬夜成了家常便饭。随着公司的技术人员慢慢增多，特别是将分布式的架构实现后，这种紧张的状况才有了好转，洪强宁开始考虑一些未来的情况，比如用户量到达一定量级时数据库的改进。做Beans DB的想法就是那时提出来的。2008年开始设计、讨论，并在2009年部署到豆瓣的部分产品中了。

当被问到为何不在旧技术上做优化，而要提出新的解决方案时，洪强宁说，技术选型必须能满足需求，否则就是白谈。当若干技术方案都能满足需求时，豆瓣会采用集体决策的方式，大家一起讨论出结论，最后授予某个人去实现。并且在复杂和简单的技术之间，豆瓣倾向于选择后者，哪怕它不完美。Beans DB的作者Davis就是一个典型，他喜欢用极端简单的东西来解决技术问题。

早期架构的分布式转变完成后，洪强宁又遇到了未曾预想的变化。最为典型的是业务模式发生了很大转变，原来是主站，发展到后来包括社区、图书、电影、音乐四个相对分开的站点，导致开发方式也发生了很大的变化，团队面临很多网站需要开发维护的情况。洪强宁坦言，如何把以往的技术积累下来，转变成可适用于多个网站的技术产品，是目前摆在自己面前最核心的问题。

出于对豆瓣整体结构透彻的了解，洪强宁逐步走向技术管理者的岗位。他强调要发挥员工的主观能动性，而不是从上面压任务，让每个员工自己去发现什么是重要的，什么是应该做的，自己则更多担负起协调者的责任，给他们提供资源。正如他所欣赏的Linus Torvolds所言。“努力寻找可以信任的人，然后让他们放手去干……一旦我让某人负责维护某个项目，此人一定会拥有处理好日常事务的能力。”

员工的情况各有不同，如何保证人人都能自觉发挥工作中的能力呢？洪强宁解释说，他们把大量的精力放在招人上面，尽量使招进来的人是靠谱、愿意主动做事的。花大力气在招人上，带来的好处是可以省掉很多管理的琐事，只要让员工了解发展方向是怎样的、目标是怎样的。至于目标怎么完成，他自己会想办法找到最好的路径。

豆瓣气质

采访中，洪强宁提到一个词，“豆瓣气质”。

豆瓣招人的时候，无论笔试、面试，更多在考察这个人，第一技术基础是不是扎实；第二对于问题思考是否敏捷；第三除了职业修养，还包括对生活本身是否有热情，是否有奉献和共享的精神。所有这些都归结成一点，是否具有“豆瓣气质”。

如何描述这样一种气质呢?

洪强宁说，在豆瓣工作的人，不是说为谋生找了一份工作，写段代码拿工资回家吃饭，而要有追求和理想。而他自己无疑是具有豆瓣气质的典型，这正是如此，洪强宁身边聚合一群热爱技术和生活的人，他们始终致力于用技术改善人们的文化和生活品质，用自己的技术口味去传播分享更好的技术，推动中国互联网的发展。

英孚教育全球CTO Enio Ohmaye

重度技术体验者

文 / 高松

孜孜不倦地用技术改善人们的学习环境，是Enio Ohmaye一贯努力的目标。在本文中，Enio也就如何看待技术趋势的价值阐释了自己的观点。

多元文化的受益者

Enio 18岁离开巴西，开始美国打工，“其实是去洗盘子、拖地板，”Enio说，“然而当时我连英文菜谱都看不懂。”

但就是这次游学经历，改变了Enio的一生。在美国，他开始学习英文、攻读自己喜欢的计算机专业，不懈的努力终于让他在美国拿到人工智能和计算机博士，并在若干年后加入Apple电脑公司。

因为能流利地讲日语、英语和葡萄牙——“还有Java, C#, Lisp, Pascal and COBOL ”，Enio会这么补充一句。Enio获得了很多工作上的机会和与人沟通上的便利。也正是出于对语言的热爱，深得多元文化涵养的好处，他决定加入英孚公司，用自己的技术专长帮助像当年自己一样的年轻人学习英语，让语言不再成为梦想的绊脚石。

1997年在英孚公司，Enio参与设计并建立了Englishtown，针对如何将认知科学与计算机技术相结合进行了大量试验，并将试验结果运用于英文教学，并通过网络实现了即时的线上语言学校。

对于自己丰富的经历，Enio Ohmaye总结自己有“日本人的身体，巴西人的心脏，美国人的头脑”。然而他未曾想到自己会来到中国，并在把自己领导的英孚全球研发中心由波士顿迁到上海。“中国团队帮助我攻关无数技术难题：SNS技术、语音识别技术、云计算和移动技术在教育行业的应用”Enio如此总结自己的中国的收获。

和Enio的交谈中，可以明显地感受到这位CTO身上有着很浓厚的苹果文化，对于自己的产品精益求精，包括产品的性能、版式， Enio说这些苹果文化对他的影响使得他在工作中更看重用户的体验。

技术让体验更美好

Enio展示了他对技术发展的观点，从最早的大型机时代，迅速到小型机时代，再到个人电脑时代。“大家都以为在下一个跨时代来临之前，微软会统治互联网桌面时代，而实际上桌面互联网时代被谷歌、百度、Facebook和Twitter等来占据，用户数量达到了10亿。在未来谁会是统治者都是未知数。”Enio说到。

Enio Ohmaye表示，科技的设备会越来越便宜，技术工作者的最大挑战就是如何把更好的体验带给用户，对于自己关注的英语学习领域，Enio认为最大的挑战是把互动式的学习体验，像网游一样带入教育行业。任何地点、任何时间、任何话题这都是学习体验最终想达到的目标。“我们最新的调研发现，20%的中国学生通过手机获得新闻和信息，我们预计这一趋势还会增长”，Enio Ohmaye补充说。

出于对未来的手机互联网时代充满了信心， Enio开始带领团队开发在iPhone、iPad上的学习应用。用热门技术。“中国会引领整个世



Enio Ohmaye表示，技术工作者的最大挑战就是如何把更好的体验给用户

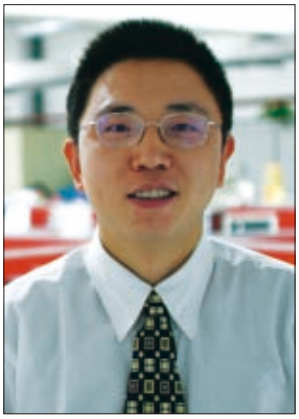
界的科技发展潮流，因为移动用户的消费习惯和数量。之前由于中国的基础设施网络不是特别好，中国相当于直接从台式机的阶段进入了移动时代。”在Enio看来，到2020年移动时代到来时，中国会成为世界上主要的使用者。

Enio指出，物联网、云计算等技术在自己团队研发的产品中都会用到，但这热点都是为了用户体验服务的，“各种样的技术都是一种工具，移动时代的到来，允许随时随地使用各类应用。”Enio表示，“未来如果有一个芯片可以制造出人的大脑进行学习的话，我们也可以做，当然核心的是我们必须切合产品模式。”以英语学习为例，我们会根据各个地区的差异性推行不同的教学模式，如小县城里，肯定会是 100%的网络教学，而大的城市里面，移动教学或许会更受欢迎。”

创新的技术手段对于Enio来说固为重要，但是如他所说技术只是手段，也是容易被复制的，“但是一个整体的体验过程是很难复制的，从基础的硬件开始到最后的学习平台，再到服务，每个环节中我们都融入了创新的服务模式，是一个完整的体验过程，这些是别人无法复制的。”Enio强调。

WWDC感想

2011年苹果全球开发者大会（WWDC 2011）于美国时间6月6日在旧金山Moscone West会议中心召开，受到全球IT从业者的广泛关注。国内众多IT人士亲赴大会现场，为我们带来了WWDC 2011中最精彩的内容。



蒋涛
CSDN &《程序员》创始人兼董事长

后PC时代进入云服务时代

2011年的苹果开发者大会（WWDC 2011）注定要史上留名，不是因为这可能是乔布斯最后一次出席，而是因为苹果首次在此大会上发布的全是软件，首次发布了清晰的云平台战略，预示着后PC时代进入云服务时代。

今年WWDC大会5200张门票（售价1599美元，合人民币1万元）在12小时内销售一空，而去年却销售了整整八天。今年5月在同一地方召开了2011年Google I/O技术大会，其门票更为抢手，5000张门票（价格100~550美元）不到一小时就销售一空。这也充分说明了开发者的新时代来临，iOS和Android是目前最热的平台。

当然Google I/O技术大会的热度多少受到去年向参会者赠送Nexus手机，今年送三星Chromebook的影响。而WWDC大会没有任何赠送，甚至连会议议程都未在网上公布，而仅仅提供了WWDC的App应用程序。但苹果开发者还是趋之若鹜，甚至好多参会者都是连续多年参加从未间断过，WWDC的最大吸引力到底是什么呢？WWDC不仅仅是苹果官方发布重要产品和技术的场所，更是苹果内部的系统工程师和开发者面对面交流的互动平台。WWDC 2011提供了100场动手实验室和120场技术讲座，出动了近千名技术工程师，我认为这才是WWDC最有价值的地方。

这次大会来自中国的参会者有70多名，其中有近20名来自高校，他们是苹果高校开发者大赛的获奖者，分别来自中山大学、浙江大学、武汉大学和中央美术学院，每年WWDC的技术大奖也有专门针对学生的奖项。其他的则主要来自中国各大互联网公司及优秀苹果App开发者。

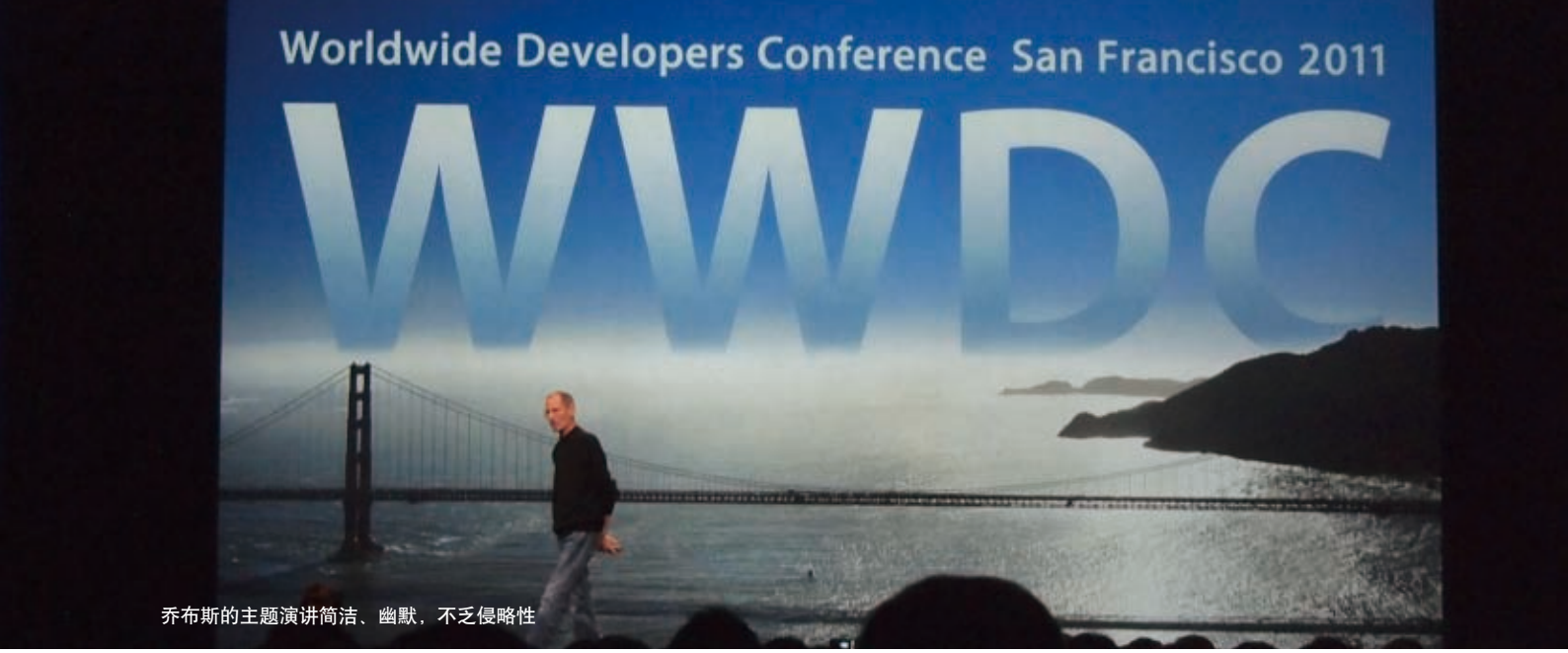
大会持续了五天，最受瞩目的是第一天上午的Keynote主题演讲，苹果发布了iCloud、Mac OS X Lion、iOS 5。后面的所有讲座都是围绕最新发布的iOS API和Mac OS API进行技术讲解，并让参会者可以现场动手，进行编程体验。

iCloud——清晰的云平台战略

苹果通过iPhone、iPad和MacBook已经在硬件设备上牢牢奠定了领先地位，下一代新产品也都在计划中陆续发布，近期其他竞争对手的产品要想超越其性价比非常困难。

要成为IT业霸主，必须构建系统平台应用的生态系统。特别在吸取了1990年代和Windows对决的经验教训后，苹果在iPhone上开创了App Store模式，聚集了50万种移动应用，成为最领先的应用平台，也极大丰富了iPhone的应用范围。

虽然苹果的软硬件平台都保持领先，但IT技术革命更新换代非常迅速，微软称霸PC时代二十年，到了互联网时代Google成了霸主，不到五年时间年轻的Facebook一举成为社交网络的王者。苹果利用iPhone和iPad成为了后PC时代的新霸主，但同时也在考虑如何在互联网服务上发力，虽然陆续推出的MobileMe和Ping都不成功，但乔布斯在WWDC2011上说完“我们学到了很多经验教训”之后，接着说了一段这样的话：“十年前我们认为PC是数字生活的中心，所有的数字内容通过PC中转。但几年前我们已经认识到，PC和iMac、iPhone和iPod一样只是一种设备，数字生活的中心枢纽是云端。”iCloud给出了非常清晰的答案，苹果系统上的所有数据，包括Email、联系人、日



乔布斯的主题演讲简洁、幽默，不乏侵略性

历、购买下载的App、照片、图书、音乐及各种文档都可以自动同步，无论你是用iPhone、iPad或iMac。只要是苹果的设备，用户无需考虑文件存储在哪里，所有这些数据之间都可以完美同步。

更重要的是，在iOS 5和Mac OS X Lion里，苹果面向开发者提供了iCloud的API，第三方应用也可以把用户数据存储在苹果的云端。原来你需要架设服务器，写好服务器端程序，并编写代码处理文件的网络读写同步。现在只需要调用iCloud的API，系统便会帮你在后台处理这一切。

相比Google Apps, Amazon的AWS，微软的Live Service，苹果在云计算服务上姗姗来迟，但苹果的风格是“Measure twice, cut once”，出手虽然晚，但一出手很猛。

从iOS API出发直接集成云端服务，开发者的第三方应用直接就变成云端服务了，“It Just Works”。对于用户来说，也无需像微软Live Mesh，Dropbox那样进行文件目录同步了。

乔布斯强调用户将不需要再有文件存储的概念，甚至也不需要存盘，就像照片一样，他只需要知道有哪些文档，而不需要了解这些文档是如何存储的。乔布斯领导的苹果曾经废掉了软驱、光驱和鼠标，现在在软件上又要废掉文件系统。

iCloud的目标是成为所有苹果用户的数据中心，包括你的个人媒体文件、工作文件、通信信息和订阅的音乐杂志报纸等。

Google、Microsoft、Amazon和Facebook也都在努力朝这个方向前进，收集所有人的信息和数据，不管你是否喜欢，我们已经进入云服务平台竞争时代。这是WWDC 2011最大的标志。

值得重视的Mac OS X Lion

另外值得重视的是，虽然乔布斯宣布后PC时代到来，但PC依然是目前最重要的IT市场，WIndows 7上市18个月宣布销售3.5亿份拷贝。

这次发布的Mac OS X Lion从iOS上吸取了很多功能，包括多点触控和Mac App Store, 新增了250个新功能和3000个API，最震撼的还是新操作系统的价格从129.99美元直降到29.99美元。

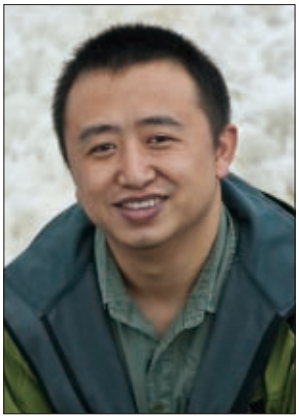
虽然Mac OS市场占有率只有5%，但借助iOS的强大影响，MacBook的销量也节节上升。拥有iPhone和iPad的用户，习惯了多点触控操作，下一步升级电脑时，Mac电脑和笔记本会对他们有更大的吸引力，这也是苹果要夺取全面胜利的关键领域。关注iOS的同时，开发者也应关注Mac OS的发展。

关于主题演讲和其他

大会开始前一天去登记的时候，有人说要听乔布斯的演讲，必须第二天凌晨就去排队。实际上当晚九点就有十来个排队的开发者，凌晨三点我到达现场时，已有近200人了。七点时已聚集了超过3000人，而当时距离乔布斯主题演讲还有三个小时。

苹果的演讲风格值得所有IT厂商学习，简洁清晰，非常流畅，不乏幽默和对竞争对手的打击。包括后面四天的技术讲座，每场讲座有两至三位工程师上场，演讲的PPT风格一致，文字和代码字体显示非常舒服，程序效果的演示也非常好。

大会特点是“Engineer to Enginner”，所有座位后面都有电源插座，现场WiFi也保持畅通。



徐达
支付宝无线事业部总经理

苹果的重要里程碑

WWDC 2011是苹果从卓越走向伟大的一个重要里程碑。本次大会上，苹果的互联网战略首次清晰化，并为进军互联网市场迈出了重要的一步。

大会上苹果推出了iCloud云服务产品，与Google、Amazon的云服务不同，它更像一个个人数据存储中心。仅仅用来存储数据，并没什么值得炫耀的，而它的独特之处在于将苹果音乐服务、系统备份、文件传输、笔记本及平板设备产品线等元素有机的结合在了一起，个人用户可轻松实现苹果各设备间的数据同步。iCloud使云计算更贴近个人，更加应用化。在我看来，iCloud是苹果由一个成功的手机制造商向伟大互联网公司转变迈出的第一步，具有革命性的意义。

相对于MoblieMe，用户每年需付99美元，即可在所有电脑（PC和Mac）上进行文件上传和下载，iCloud则完全免费使用，同时仅限于苹果设备。这将直接导致苹果可以拥有用户的所有数据，借此更好掌握用户的习惯及偏好。可以预见，未来iCloud也只会对苹果用户开放，用户对苹果设备的粘性将被增强。因为用户数据放在苹果的云端，今后其换用其他设备的成本会比以往更高。

除了iCloud，大会给我带来的另一个震撼就是Mac OS X Lion的iOS化。Lion中增加了很多iOS元素，使其在用户体验上与其他iOS设备更加贴近，可用性更好。借鉴App Store，Lion内置有Mac App Store。用户可以通过Mac App Store，实现Lion及相关软件的空中购买、安装和升级，实现一次购买、永久免费、随意安装，从而使用户的忠诚度和粘度更高。可以说Lion给用户带来的体验已远远超过了Windows 7及其他桌面操作系统。

至于iOS 5，除了重大发布“通知中心”外，其与Twitter的整合，也引起了大家的关注。现在苹果手中掌握着大量的用户数据，他们希望构建一个既有设备平台又有软件服务的、关系更强的社交网络，所谓关系更强是指把原来用户之间的关系变成用户和苹果间的关系，从而弱化用户间的关系。而Twitter是一个与用户关系更强的社交网络。依靠Twitter的植入，苹果可以把用户体验控制得更好，从而可以直接与Facebook相抗衡、竞争。

本次大会并未发布新一代iPhone手机，或许让有所期望的人稍感失望。但我觉得会上推出的所有软件，对苹果来说，意义更重大。正如上文提到的，在大会中苹果的互联网战略首次清晰化。只要苹果坚持此方向发展，即便将来乔布斯离开了公司，苹果仍旧可以做到卓越。



彦风
中央美术学院设计学院讲师、苹果中国杰出教育工作者

设计师眼中的 WWDC：更开放、更互联

这次去参加WWDC 2011，缘因今年中央美术学院在苹果中国高校学生开发作品大赛中获得两个大奖：最佳用户体验评委会特别奖和三等奖，从而获得了本次大会的邀请，这也是我第二次参加。

这次参会首个深刻体会就是参会人数明显增多，总人数达四五千人之多。参会者包括苹果粉丝、开发者和参会会员，甚至还有同行的竞争者。“到处排队”现象足见参会者对此大会的热衷。在我看来，iPhone、iPad的推出之后，不仅吸引了众多应用开发团队参与到这个平台上来，也带动了一批衍生品的产生，如iPhone外壳、iPad支架等，苹果现在是一种潮流，最后将会变成一种文化。

大会首日，能一睹“乔帮主”风采，不可否认，是大部分参会者此行的目的之一，虽然乔布斯更加清瘦，但听到他带来的Keynote演讲，着实让大家兴奋了一把。透过乔布斯现场发布的iCloud、Mac OS X Lion、iOS 5三大产品，我们可以看出他正开始改变以往保守的心

态——Lion从129.99美元直降到29.99美元；iCloud提供音乐共享，并可同步到众多设备上，以及其他免费的产品，就是很好的证明。现在苹果正为开发者提供更大的操作空间，用更大的怀抱来迎接开发者，这也是对抗强势推进的Android开发阵营的一个策略。

大量的用户群及迅速增长的需求，决定了移动设备将是未来发展的一个主要方向。设备间的互操作现在还是一块空白，苹果优先满足了这个不存在竞争的区域——在大会上重磅推出iCloud，实现苹果各设备间的互操作，用户体验更加简单。乔布斯在演讲中指出“我们将把PC和Mac降格为普通设备”，这也预示着iCloud将取代iTunes成为新的数字中枢。iPhone、Mac等终端设备只是人们享受生活、工作的工具而已，终端设备上的互联互动、体验丰富的服务才是苹果将要关注的核心业务所在。以后不排除iPhone会有免费的可能，苹果未来也将更开放。

透过WWDC大会，可以看出苹果做事很专业、很科学。无论大会的举办、产品的发布，一切都做得平平稳稳，但又总能有激动人心的兴奋点。这是国内一些团队、机构甚至联盟所应该学习和借鉴的地方。同时我们也应该好好思考一下我们未来产品的定位及今后要走的方向，外围产品建设也不容忽视，这是本次大会给我带来的又一感触。



本届WWDC吸引了超过5000名的参会者



微博热议 WWDC2011

iOS 5“借鉴”了其他几乎所有主流操作系统的优良传统，另一方面，部分开发者也被“拿来主义”，不少优秀的App在本次大会之后估计要销声匿迹了，本次抄袭大会名副其实。

——手机之家执行总编葛振宇

Apple并不是与开发者争利，他们在考虑是否增加这些新功能时，根本不会考虑开发者的那点收入，更是毫不在乎的让其全部免费；Apple也不是在模仿Android，他们只是关心如何通过硬软件提供一个出色、用户体验好的产品，至于合作伙伴都是次要的。可以遇见，未来就是新的一个“傲慢帝国”。

——TMT行业专家高楼壁虎

在WWDC2011上，苹果发布的iOS5中直接包含iMessage，很多人问我，米聊怎么办？我的观点是，连苹果都要干，这说明米聊大方向是对的，大势所趋。米聊是否能胜出，就看看米聊是不是比iMessage做得更好。我们不会服输的，米聊将会发布新的大招，大家等着。

——著名天使投资人雷军

Mac OSX Lion给我的感觉有：1.iOS体验反攻到电脑；2.新的用户体验，仍然有部分依赖于新的Hardware（Track Pad），Apple的垂直整合策略发挥作用；3.Apple在关注的一些用户体验，AutoSave、Versions等，微软总是没做好；4.PC工业仍然有很大增长潜力，并不像神话中说的今后都是互联网终端。

——移动立方创始人黄晓东

iOS内置直接覆盖所有用户，再好的APP能覆盖10%用户已经很牛逼了，怎么比？苹果可以免费，第三方要靠这个挣钱，怎么打？有些API不开放，第三方开发受限制，怎么玩？

——篱笆网联合创始人徐湘涛

说苹果进入开发者地盘的人忘了，苹果还有几大杀招没出手：语音输入、地图及导航、地理位置相关服务、基于手机通讯录的交友和分享、手机钱包……

——创新工场投资经理张亮

云是一种潮流，苹果也在WWDC2011上推出了iCloud，但使用方法与亚马逊和谷歌不同，选择不跟风。一向引领风尚的苹果“嗖”到了什么？了解内幕的人都不得不佩服苹果惊人的前瞻性。

——麻省理工科技创业

iCloud会是一个划时代的产品，秉承了Apple产品一贯的原则，越复杂越需要简单。其他各家的Cloud马上也会开始学习的。

——北京市活力天汇科技有限公司产品事业部总经理史册伟

【WWDC大会告诉了我们什么？】我写的一篇很长的观察文章，发到CSDN了，欢迎大家指教：http://t.cn/aokWpJ。主要观点：1.iCloud是苹果全面拥抱云计算的开始，云是计算机，数据是竞争要点；2.移动优先成大趋势；3.腾讯化是大势所趋；4.苹果与Twitter结盟：合纵连横时代开始。

——CSDN&《程序员》杂志总编刘江

苹果云服务的目的是把所有的iOS设备和PC进行一次文件变革。文件是历史上我们最习惯的管理模式，而苹果现在把所有操作中的存储操作取消了，代之以类似Google Docs的版本管理和自动存储。让你在任何设备上，都像在使用一个设备。

——中国企业家杂志

在苹果的愿景中，云的作用是让本地应用更好地发挥作用，苹果云属于“存储转发”云，你的Apple设备是宇宙的中心；而谷歌则把云视为本地应用的替代者，视网络为宇宙的中心。此二者能否同时正确呢？

——视讯中国战略合作中心总监那鑫

Mobile Development

TOP20

值得推荐的20家移动应用团队

策划 / 本刊编辑部

移动互联网的热潮下，创业团队如雨后春笋涌现，但大浪淘沙，真正脱颖而出并非易事，根据近期CSDN网站“2011年中国软件开发者有奖大调查”的最新统计数据显示，目前真正实现盈利的移动创业团队仅占25%左右。

移动创业者们该如何脱困？应用的立项、设计和研发、运营和推广这些环节均是事关一款应用能否成功的瓶颈，为此《程序员》特别在国内的诸多创业团队中，筛选出了“值得推荐的20家移动应用团队”，他们均在不同层面取得了成功，并在上述环节中都有其独到之处。

根据我们对20家团队的进一步追踪采访发现，绝大多数均是最近三年内创立的，不得不让人感叹一句老话“时势造英雄”，当然也有部分团队，如顽石互动、掌上明珠等的核心创始成员，在移动领域有接近10年的积累，但他们对移动产业、个人奋斗的价值有着令人耳目一新的理解，我们也对他们做了详尽的专访，其心路历程，其经验教训，相信能引起您的共鸣。由于篇幅所限，本期杂志不能刊出这20家优秀团队的所有采访内容，未发表的部分，我们将在CSDN网站陆续推出，欢迎关注。

20家值得推荐的移动应用团队（注：各类别的团队排名不分先后）

类别	企业/团队名	创立时间	核心产品
游戏	顽石互动(北京)网络科技有限公司	2006年	游戏《二战风云》
	掌上明珠信息技术有限公司	2003年	游戏《明珠三国》
	成都优聚软件有限责任公司	2009年	游戏《帝国塔防》
	北京捷通华声语音技术有限公司	2000年	游戏《天行乐园》
金融、电子商务	金蝶理财网	2010年	金蝶随手记
	上海益实多电子商务有限公司	2008年	掌上1号店
	国信证券股份有限公司	1994年	金太阳手机证券
	九九房（北京万瑞松林网络技术有限公司）	2011年	看房日记、掌上买房、掌上租房、房产资讯
社交	北京推图科技有限公司	2011年	推图
	深圳市嘀咕网科技有限公司	2010年	嘀咕2011六月版
	珠海市伟衡科技有限公司	2011年	圈圈
	北京沃安科技有限公司	2009年	“炫拍”-手机视频分享
	北京木瓜移动科技有限公司	2008 年	Papaya Mobile 移动社交平台
平台、工具	Splashtop	2006年	Splashtop Remote
	AdView	2010年	AdView移动广告优化工具
	中国电信股份有限公司应用商店运营中心	2010年	天翼空间
	重庆掌脉科技有限公司	2010年	掌脉智能手机跨平台阅读系统
	北京千尺无限软件技术有限公司	2008年	千尺下载
	北京爱普之星科技有限公司（app梦工厂）	2010年	App123
	安徽科大讯飞信息科技股份有限公司	1999年	讯飞语音输入法

顽石互动创始人吴刚：走高品质的发展道路

口述 / 吴刚　整理 / 杨东杰

从1997年开始，吴刚已在游戏领域从业了十几年，出品了数十款游戏产品，两次成功的创业经历更是积累了从游戏制作、运营到团队建设丰富的经验。近日，《程序员》记者专访吴刚，并将这些心得整理成文。

我在15岁的时候就疯狂地痴迷上了计算机编程，现在还记得当时的投入劲头，经常通宵达旦地坐在电脑面前，以至于骑车回家的路上困得睁不开眼。通过一段时间的学习，我还创作了一个图形化编程工具，叫做BoyWindows，这款软件为我带来数十万元的收入。

当时的我就已经触摸到自己的理想——能够做自己喜欢的计算机工作，通过它解决别人的问题，让自己赚到钱。所以高中毕业后我没有上大学，直接进入IT行业。

第一款游戏的血泪教训：发誓再不做烂游戏

由于第一款软件的成功，1996年，年轻气盛的我在一个老板的资助下，开始了我的第一款游戏项目，这是一款叫做《血狮》的单机游戏，投入几个月的开发，出来的产品却非常糟糕，上市后，接到来自全国各地大量用户的退货。我由此深刻地感受到产品品质对用户的重要性，发誓再不做烂产品，即使在疯狂的SP时代，身边的很多人做垃圾游戏赚了很多钱，我都能坚持住——我做一款烂游戏就够了。

《血狮》带给我的不仅是血泪教训，也让我感受到游戏的魅力。于是2000年，我启动了第一家自己的公司——数位红。公司接到的首个项目来自诺基亚，当时诺基亚9210采用了首个开放性操作系统EPOC（Symbian的前身），



吴刚认为，只有热爱生活，才能做出好游戏

希望我们做一些运行于诺基亚9210上的游戏。

数位红刚刚起步，资金和人手都非常短缺，但钱少有钱少的做法，我先花了几千元让一个朋友帮忙开发出游戏引擎，然后基于这个引擎，找到几个兼职开发者按照我的策划开发游戏。由于是分阶段付款，所以几万元就可以实现几个项目一起启动。

把游戏提交给诺基亚后还遇到一个波折，诺基亚方面不能按照合同收购我的游戏，后来经过谈判，答应将游戏推荐到海外的渠道销

售。由于当时诺基亚在海外的游戏供应商非常少，我就因祸得福成为诺基亚Symbian平台上最主要的手机游戏提供商，从那时起开始主要做海外市场。

数位红后来发展得越来越好，摩托罗拉、索爱、NEC等主流厂商都成为我们的合作伙伴，游戏被大量预装到手机上，到2004年9月被盛大收购时，已经是国内最好的手机游戏厂商了。

再次创业：走出一条高品质的发展之路

数位红被收购后，我赚到的钱已可保证有体面的生活，但我还是非常喜欢做产品，喜欢和一群小伙子们一起打造出一款款优秀游戏的感觉，另外，我希望通过成立公司使我的团队成员也同样过上衣食无忧的生活。

所以在2006年末我二次创业成立顽石互动，虽然对游戏产品的整套制作理念在数位红时期已经基本成型了，比如主打海外市场，偏向手机平台，踏踏实实做精品游戏等，但新公司的发展理念同第一次创业相比更加清晰，我希望顽石走出一条以高品质而不是以规模为优势的创业道路。高品质的要求不仅在于产品和服务，也包括公司的运营情况和员工待遇。

顽石互动发展到现在，包含客服人员仅有一百多人，但是，我们的利润率远远超过同行，去年我们的收入是四千多万元，今年的目标约两个亿人民币。尽管发展很快，但我们的人员速度扩张却不快，因为我并不认为员工规模和公司发展呈正比关系，相反，很可能是员工越多，公司对员工待遇的提升越缓慢，员工对公司的满意度会越差。同样，上市也不在公司发展计划内，我只想管好我的一百多号人，并不想为成千上万的股民和一堆董事负责。当你把盈利、长期持续增长作为公司目标时，给员工的回报会非常可观，而我们正在迈进这一阶段。

而这一切无不以推出优秀并受到市场欢迎的产品为基础。顽石互动现在每年平均推出一两款产品，每款都精心打造并获得可观的盈利，团队也保持了很好的稳定性。今年我们主打的《二战风云》近日还登上中国iPad游戏收入总榜的第一名。这个产品在全球四五十个国家策略游戏类别中也都进入了TOP10。

游戏成功第一关：选对题材和类型

在游戏行业摸爬滚打十几年，基于对游戏下载、安装、注册和过关等用户数据的积累，我总结出一个判断游戏市场的“三三”理论：游戏前三秒主要表现在题材和视觉体验。首先用户会在自己喜欢的题材分类下寻找好的游戏，比如武侠类还是战争类。用户下载安装后，从启动画面、菜单分布到游戏主界面是否能给用户强烈的吸引力，是否感受到游戏制作精良——70%的用户在游戏的前三秒钟内决定是否玩下去。

上手三分种内，用户会判断游戏类型（如即时战略、对战类）是否投自己的胃口，操作环境和菜单是否符合用户习惯、交互是否简洁流畅。而前三小时则考验用户对游戏的世界观、核心玩法和任务设计等是否认同。能继续玩下去的玩家里，90%是该游戏的真正用户群。

所以打造一款受欢迎的游戏最重要的一步是选择游戏题材和类型。我平常会花很长时间来分析热点和趋势，寻找空白市场。比如大家都认为在Symbian上做网游没有市场，很长时间里没有一款针对Symbian平台设计的优秀网游产品，都是一些J2ME版本的移植游戏。我们就在一年半前推出了《契约OL》，并取得了很好的成绩。而《二战风云》针对的空白点是现代战争题材。在PC游戏市场里，每年的热门游戏名单中都有现代战争题材类游戏的位置，更是涌现出不少像《荣誉勋章》、《使命召唤》这样的经典游戏，但在手机网游这个领域中，却没有几款同类题材的游戏。

主要面向利润丰厚、成熟良性的海外市场

由于我们面对的是全球用户，必须选择大多数国家都有参与的战争事件，排除掉十字军东征、拿破仑等偏欧洲主题，只有二战最为合适。《二战风云》目前的成绩也证明了我们决

策的正确。

顽石互动的用户和收入超过三分之二来自海外，国内的市场只是借势来做。主要原因是国内不成熟的市场环境更适合更新节奏快的产品，这类产品一般会频率很快地推出活动，以增大ARPU值（每用户平均收入），但也缩短了产品生命周期。相比之下，海外市场更加成熟，节奏较慢，也更加符合游戏的本质——享受乐趣而不是急于追赶。一旦在海外市场站稳脚跟之后，能维持相当长时间的良好发展，而不是需要像国内要做很多偏离本质的推广工作。

我们相信高品质一定伴随着专注和耐心。一般和海外联运商建立合作后，我们会持续稳定地投入，并不要求马上盈利，可能刚开始产品的表现并不是很好，但经过几个月的调整和改进，收入都会慢慢上升。更重要的是，这款游戏的团队会因此获益良多，他们对于产品的改进和用户需求的理解是持续积累而不是浮在表面。

坚持跨平台策略

一个平台就意味着一部分用户市场，我们对跨平台的尝试远远走在同行的前面。原先，顽石互动被认为是手机游戏提供商，但现在看来，我们已经成为跨平台的游戏提供商。你可以看到，我们公司只有两款手游产品，但在主流平台上都有相应版本，而且各平台均是盈利的。

跨平台特性在游戏最初设计时就被引入，但我们总是以某个特定平台为切入点，先在第一个平台上赚到钱，才扩展到第二个平台。平台扩展顺序主要根据产品类型和该平台市场规模及竞争对手情况而灵活制定，比如《二战风云》最先发布的是Web版本，然后是Java版、WAP版、MTK版，最近才扩展到iOS和Android平台。而《契约OL》，我们是先选择Symbian平台切入，然后再扩展到iOS平台，再到Web。基本上顺序和同行们的习惯相反，这也有团队在不同方向快速扩展的考虑。

而且，不同于很多公司以平台划分团队，比如Android团队负责公司的所有产品在Android上的开发，在我们顽石内部，是以产品划分团队的，《二战风云》的所有版本都由这一个团队来做，这样做尽管牺牲了扩展速度，却可以保持产品品质的稳定性，而且对团队深入了解各平台的特性帮助也很大。

我的角色是产品制作人和创新推动者

在公司，尽管我的头衔是CEO，但公司事务性工作管得很少，更多的时间是和研发团队在一起。我的角色主要是两个，第一个是产品制作人，分析市场趋势、主导产品方向以及预算审批等。公司的每个产品在原型确定前都是我来主导，Demo确定后我就脱离了。

我非常讨厌软件工程之类的功能模块化研发组织模式，更青睐采用原型迭代的方法，就像画家创作一幅油画，总是从素描轮廓开始逐渐细化。在产品的最初阶段，我会特别强调做Demo，如果在最初的Demo上也能看到鲜明的游戏性，那这款游戏的成功前景就会很好。

我的第二个角色是负责公司的创新实验室。实验室拥有一支近20人的创新团队，研究各种新技术和平台，尝试开创性的产品理念等。这个实验室还有培养新人的功能，可以在不压抑员工创新思想的前提下，将我在游戏行业十几年的经验教给新人。游戏作为创意行业，最不应该存在的就是权威，所以在公司内我非常鼓励员工“三拳打倒老师傅”。

创业以来，在人才培养和团队建设上，我花费了很多精力。我们公司的人才流失率非常低，而且顽石一直保持着从来不向同行挖墙角的原则，原因就在于我们更愿意每年花大量人力、物力来培养大批优秀的新人，从程序员到策划均是如此。

很多人认为创业是个艰苦奋斗的过程，创业者必须兢兢业业，我并不赞同。尽管每天的工作时间经常超过10~12个小时，但我会每年集中休假一个月，完全与工作隔离，去痛快地旅行，或者投入到喜爱的户外或赛车运动中。我认为创业是种生活态度、生活方式，游戏创作者的文化内涵、价值观、生活品质在某种程度上是游戏成功的决定因素之一。只有热爱生活，才能做出好的游戏。

木瓜移动：从平台到孵化器
——木瓜移动创始人沈思专访

记者 / 杨鹏飞

木瓜移动（Papaya Mobile）是一个开放的移动社交平台，专注于Android平台的休闲游戏和虚拟货币，其公司团队成立于2008年。创始人沈思本科毕业于清华，并获得美国斯坦福大学计算机及管理科学与工程双硕士学位，曾负责美国Google总部重要项目的开发和管理。木瓜移动在全世界拥有1500万注册用户，并以每日5万新增用户的速度增长。

木瓜游戏引擎是一个基于Android平台的免费OpenGL游戏引擎，让开发者能在iOS和Android平台上开发较复杂的社交游戏。据报道，木瓜移动现在已经开始进行孵化器计划，为游戏开发团队提供支持，无疑这是一次重大的战略转型，他们是如何理解中国的游戏开发、是如何考量这一新举措的？本刊记者采访了木瓜移动创始人沈思。

记者：木瓜移动的最初目标是什么？开发人员所占比重有多少？

沈思：木瓜移动成立于2008年3月，最初的目标就是做一款“移动互联网上的游戏社交平台”。做平台的话就要先构建起自己的生态链，而作为一个游戏社交平台，想要支撑上面的游戏的话，需要有一个跨平台的游戏引擎。所以我们就先做了一个游戏引擎，然后在这个引擎的基础上，通过自己开发的一些游戏，构建起整个社区的基础。

我们在北京总部有60多人，美国旧金山有10人左右，成都和纽约也有一些员工。而这些员工里面，和开发相关的人员占到80%。

记者：当初为什么会想到木瓜移动这个名字？

沈思：公司的名字是和我的经历有关的。我在美国的时候非常崇拜苹果公司的Jobs，有一次我去听Jobs的演讲，之后便萌生了要创业的念头。后来成立了木瓜移动，在起名的时候也是希望能起一个和苹果相关的名字，于是就在所有水果的名字中找，最后选定了木瓜这个名字。因为木瓜的英文叫做Papaya，念起来也比较朗朗上口一些。

记者：你们一开始就是针对Android平台进行开发吗？

沈思：从我们公司的历史来讲，最早是做J2ME方向的。后来苹果公司发布iPhone之后，我们成为了首批的iOS开发者之一。但是因为苹果App Store过于严厉的政策，并不利于我们平台的发展。于是在经过思考之后，将战略重心放在了Android平台，其相对宽松的政策对我们的发展更加有利。不过我们现在也在进行iOS平台的开发，相信不久就可以移植到iOS平台上。

记者：现在提供的游戏有多少？第三方合作伙伴中，其中小团队的开发者和大公司之间的比例是多少？小团队的优势在什么地方？

沈思：现在木瓜移动上有200多款游戏。其中有10多款是由木瓜移动自己开发，但是我们已经不再进行游戏开发了，更多的是帮助第三方合作伙伴进行游戏开发。在我们的平台上小团队和大公司之间的比例大概是1:1。

大公司因为有战略上的考虑，需要做一些长期的规划。而小团队相对于大公司来说更加灵活，执行力比较高，可以快速地开发出一款游戏，迅速抢占大公司没有进入的领域。而且如果给予一定的帮助，小团队开发出来的游戏质量不会比大公司差。

记者：你们的平台带给开发者的便利是什么？

使用你们的SDK会有什么样的好处？

沈思：我们和其他的游戏引擎、平台选择的策略不同。我们先在不同的平台上搭建一个自己的脚本语言的开发环境，所有的游戏开发都使用统一的脚本语言来实现，从而对开发者降低了开发门槛。开发者不需要掌握复杂的Java语言或者C++语言，只要会用一些脚本语言，就可以利用我们的SDK轻松实现游戏开发，目前我们主要支持的脚本语言是Python和Flash脚本语言。

游戏引擎里实现了繁琐流程的处理，如OpenGL渲染、连接Java和C、网络传输、本地cache等。开发者不用重新实现这些模块，这样一来带给开发者的好处就是，开发者只需要专注于游戏的策划和程序功能的实现，而不用将精力浪费在平台移植和屏幕分辨率适配等问题上，同时省去了学习Java或者C++的步骤。

记者：这样一来，程序的性能和运行的效率是不是受到影响呢？

沈思：木瓜游戏引擎是基于Android平台的免费OpenGL游戏引擎，引擎核心是用C++实现的。其实一个游戏引擎最重要的是如何更好地利用设备底层图形处理能力，事实上该游戏引擎相对而言是更占优势的，因此语言对程序运行效率的影响就相对较小。

记者：你们怎么看国内市场？是如何对开发者的游戏进行推广的？

沈思：我们认为用户在移动终端上的社交需求正在迅猛增长，在中国这个趋势非常明显，移动现在已经是非常热门的领域，移动社区将成为移动互联网的基础应用。而移动社区已经成为中国用户的最迫切需求，中国市场潜力巨大。手机游戏是中国市场最好的应用，“手机游戏+移动社区=手机游戏社区”已成为最佳组合。推广方面，我们在游戏里提供了在游戏内支付的功能，用户可以直接在游戏中进行交易。

记者：你如何看待木瓜移动的未来？

沈思：木瓜移动在这么多年的运营下，已积累了相当丰富的经验。不光是在游戏引擎的开发上，更在于其他方面，比如社区运营、商业模式、应用推广、平台建设等方面都有很好的沉淀。我们的目标是成为移动互联网上最大的游戏社区，当然从用户数量上，现在已经是Android上最大的游戏社区了。我们现在更多的是以扶持整个产业链的态度来做事情，而在游戏孵化这块，目前国内还是空白。

现在，手机游戏开发的门槛现在已经很低，一般几个人就可以成立一支团队。国内很多开发者都是大学毕业生，在游戏开发、商业推广等方面的经验相对比较缺乏。木瓜移动现在开始做一个孵化器，给他们提供硬件、软件上的支持，包括各种各样的培训和商业推广。从游戏的开发开始，到后期的推广和策划，到最后游戏的盈利，进行全程的支持。



沈思认为，目前手机游戏开发的门槛已经很低

手机网游是移动应用的第一个热潮
——掌上明珠创始人武春雷专访

记者 / 长生

掌上明珠，是中国本土最为成功的手机网游公司之一，从2003年初创时的20多人，到现在近400人的团队，掌上明珠在移动互联网的大潮里，取得了飞速增长。8年间他们开发了近百款手机游戏及应用，积累了无数值得业界借鉴的经验。为此本刊记者采访了掌上明珠创始人武春雷先生，请他分享在移动大时代里做手机网游的成功心得。


武春雷认为，手机网游是移动应用的第一个热潮

手机网游的机遇

记者：请谈谈创立掌上明珠的背景故事。

武春雷：我们是在2003年成立掌上明珠的，有4个创始人。之前我们曾经创立一家公司，战略目标定在外包软件上。但2003年时，我们发现给韩国、国内SP做的外包工作中，来自无线领域的业务迅速增长，便意识到在移动互联网领域内一定会有伟大的软件公司出现，奔着做“移动互联网领域的MicroSoft”这么一个目标，我们成立了掌上明珠。

刚开始，我们还是做无线领域的外包工作，随后进行了证券系统、邮件系统、浏览器、手机阅读等诸多方向的探索。2005年时，我们正式发布了自己的第一款手机网游。

记者：为什么后来觉得手机网游更加有前途？

武春雷：我们在研究新科技对人类生活带来的影响时，发现是一个浪潮推动一个浪潮向前走的。不同的时期，会有不同类型的产品崭露头角。而我们以前探索的手机邮件、浏览器等应用，在我们创业那会儿，发展的时机并没有到。因为当时彩屏手机非常少，即使市面上已经有诺基亚401，可是屏幕就那么一小块，对于那些应用来说，总有一点削足适履的感觉。

与此同时，我们在游戏领域做了一个很简单的尝试，一位工程师大概用了两三天的时间，做了一款开心扫雷，相当于把PC上的扫雷游戏直接搬到了手机上，然后放到了移动梦网的百宝箱里提供下载。我们平常的应用，如EMAIL系统、个人信息系统等，每天的下载量基本上是个位数，而开心扫雷每天的下载量超过了一百个。这对我们来说是一个惊喜，并意识到在移动互联网产业内，能够首先成功的应该是游戏。于是掌上明珠正式开始游戏方面的尝试。期间，我们一直考虑这么一个问题：为什么PC单机游戏并没有成功，成功的全是网络游戏？我们认为在手机领域，同样是网游最有前途。于是2005年，我们基于之前的RPG游戏引擎，以及无线通讯领域的技术积累，开发了第一款手机网络游戏《武林擂》。

掌上明珠的里程碑

记者：从2005年到现在，掌上明珠的网络游戏，经历了哪些里程碑意义的阶段？

武春雷：掌上明珠这么多年来，几乎每一款游戏都意义重大。我们的第一款游戏，往往爱说成不是网游，而是第一款单机游戏《悟空传》。它奠定了我们整个游戏引擎最底层的一些基础，因为掌上明珠的创始团队成员，全是技术背景。这款游戏基本把我们自己对游戏的理解贯彻进去了。

《武林擂》是我们向网络游戏迈进的第一步，它证明了在手机上可做网游，我们的画面表现、内容是玩家可以接受的。在这个前提下，我们进一步开始探索手机网游到底能够做到什么样子，于是有了第二款网游《幻想i时代》，它几乎把所有的手机网游元素全部加进去了，有装备、宠物，还有如师徒系统、婚姻系统等，甚至包括小游戏——几乎所有的内容都有。

《明珠三国》实现了手机网游质的飞跃，它不仅是手机网游界三国题材中首款Q版ARPG即时战斗大作，更是首款跨全平台手机网游。在这款游戏中，我们设定了很多独创的玩法，同时也在游戏画面上实现了小客户端、低流量同时画质清晰、动态展示逼真的效果。

做有移动应用特色的网游

记者：有人认为，手机游戏的玩家往往正处在诸如坐地铁这样的碎片化时间里，所以需要投入大量精力的网游不适合手机，您怎么看？

武春雷：其实，我们一开始做网游，就在考虑如何充分地利用手机碎片时间这个问题。《武林擂》有一个擂台战斗系统。打一个擂台，从开始的淘汰赛，到最后的循环赛、冠亚军争霸赛，这个时间是非常长的，但需要用户一直在线实际上是不合理的。所以我们采取了这样的策略：擂台的比赛是按照武打套路走的，玩家可以提前把自己使用的套路提交给服务器，这样即使玩家不在线，比赛也可以照常进行，等玩家上线后可以看到比赛结果和回放。再比如副本设置，在PC端上诸如《魔兽世界》等游戏，副本任务一般会需要两小时，这个时长放在手机上就是不合适的。所以对于《明珠三国》，副本的一般设定会在10分钟以内。对于那些上线时间极短的玩家，我们也提供了相应的服务元素，比如“龙盘系统”，有点像老虎机，当玩家把装备放进一个系统提供的轮转盘，就有机会赢得更高级别的装备，这也是为了让玩家即使短时上线也同样能够充分享受游戏乐趣。

此外，根据我们多年运营游戏的经验，发现玩家其实也不总是都利用碎片时间玩手机游戏，我们的用户平均每天在线时间达3小时，这和PC网游非常接近。

记者：针对手机用户的操作特性，你们在游戏中做了哪些独特设置？

武春雷：我们一直在摸索玩家该怎么操作才最方便。就以前的按键手机为例，最重要的就是几个按键。我们有这样的理念，当一新手进入网游不知道该干什么的时候，下意识中会按的那一个键，是被我们重点关注的。比如键盘中间的“5”键和“OK”键有等同的功能。另外，很多手机是没有“上下左右键”的，所以我们将键盘上的“2、4、6、8”充当上、下、左、右的操作。剩下的1键、3键、7键、9键、*、0、#，则作为一些辅助的快捷键，比如启动聊天功能等。另外，我们还创建了类似于密码的数字菜单，即玩家只需输入几个数字就可启动菜单中所需的功能，不仅操作方便，也有利于玩家与客服、玩家之间的沟通。当然，现在触屏时代已经来临，我们的产品也正在做相应的革新。

记者：从游戏制作效果而言，PC网游更有优势，手机网游该如何吸引PC网游的玩家？

武春雷：首先，要靠游戏本身去打动玩家，比如它是否煽情、是否迎合玩家的世界观，是否能给玩家带来意外惊喜等，这需要在制作方面下苦功。其次，我们努力创造一个促使玩家之间进行良好互动的环境。我们曾经有句玩笑话，玩我们的手机网游可以省钱，因为玩家如果大量靠短消息交流，时间一长短信费用就哗哗地流出去了。而在游戏里，则可以密聊、可以飞鸽传书，当然也可以公聊展示自己。除了缴纳游戏的流量费外，内容本身是免费的，所以客观上反而省钱了。

移动应用优秀团队推荐

splashtop

创立时间：2006年

核心产品名称：Splashtop Remote

产品简介：让用户通过移动设备（包括智能手机和平板电脑）遥控Windows PC或者Mac，在移动设备上获得完整的PC使用体验。

适用平台：Windows、iOS、Android、MeeGo

运营成绩：从2011年5月30日到现在，Splashtop Remote在美国的App Store里一直是iPad付费应用下载排名的榜首，并且在其他二十几个国家/地区获得过第一的位置。在Top Grossing iPad App排行榜里面，该应用排名第二。

创始人简介：Mark Lee，Splashtop共同创始人兼首席执行官。在成立DeviceVM之前，Mark是OSA Technologies公司的共同创始人和首席执行官。OSA公司在2004年被Avocent以1亿美元并购。Mark作为Avocent的高级副总裁续任至2006年6月。

产品特色：在一些指标性测试中，Splashtop Remote Desktop能够以稳定的每秒30帧的速率播放视频，而且只有相当低的延迟时间，而几大知名竞争对手的产品则是以每秒平均2到6格的画面播放，与之相比低了80%到95%，这样使得影片的播放效果更像是看幻灯片。从延迟的数据来看，这些产品的平均延迟时间要比Splashtop Remote多出2~10倍，用户基本无法顺畅操作。

成功建议：应用本身有前瞻性和良好的发展前景；在某一点上做到最好，竞争对手完全无法复制。

九九房

创立时间：2011年初

核心产品名称：看房日记、掌上买房、掌上租房、房产资讯

适用平台：iOS、Android

运营成绩：用户量25万左右，下载量60万左右，活跃用户数10万左右。

创始人简介：张一鸣，2005年毕业于南开大学软件工程专业，曾任酷讯网技术委员会主席，组建并管理最核心的搜索研发团队。

成功建议：成为自己产品的忠实用户，了解用户的真实需求， 重视用户反馈，不断尝试和快速迭代，努力把产品做到极致。

北京爱普之星科技有限公司

创立时间：2010年

核心产品名称：App123

产品简介：App123是一款高效的导航软件，可通过筛选迅速定位需要的软件，并可查看内容提要、截图预览、排名、评分及评论，为用户推荐iPad/iPhone装机必备软件。

适用平台：iOS

运营成绩：月独立用户量：12万；下载量：29万；活跃用户数：1.2万/天。

创始人简介：杨大川，现任app梦工厂CEO。1997年毕业于美国纽约州立大学布法罗分校，计算机硕士。1999年至2002年任美国硅谷Annuncio公司（后来并入Oracle）首席工程师；2003年参与创办思奇联盟，旗下包括迈思奇、艾德思奇两家高科技公司，2010年创办app梦工厂。

成功建议：做一个成功的移动产品，无论从功能还是界面上都要尽可能地考虑用户的习惯：良好的用户体验是产品成功的第一步，而从功能上，增加必要的功能，减少可有可无的需求，尽可能地使用户操作更方便、更便捷。当然功能之外更重要的就是性能和速度，如果一个产品功能再好，速度很慢也不会得到用户的青睐。

GOODTEAM STUDIO 优聚软件

创立时间：2009年4月

核心产品名称：帝国塔防

产品简介：一款以帝国时代为背景的塔防类游戏，配备新的魔塔和4张地图，可以通过游戏中的水晶去解开更先进的魔塔和新地图，才能战胜愈来愈强大的敌人。

适用平台：iOS、Android、Symbian、BlackBerry

运营成绩：下载量80万，用户量80万；活跃用户数32万

产品特色：塔防类游戏在Web和手机上都很受用户欢迎，同题材产品层出不穷。帝国塔防的产品特色在于：1. 选择游戏玩家都非常熟悉的帝国时代为设计背景；2. 同类游戏多以固定线路居多，独特地采用自由地图设计，让玩家可以自由的发挥和设计线路；3. 增加商店、成就系统，让玩家除了地图的线路设计以外也增加了可玩性；4. 在美术设计和Android多平台多版本支持上投入了很大精力，支持几十种机型的Android手机。

深圳市嘀咕网科技有限公司

创立时间：嘀咕网隶属于中友集团，中友集团成立于2004年。2010年7月，嘀咕网正式推出手机嘀咕，转型提供位置社交服务。

核心产品名称：嘀咕2011六月版

产品简介：嘀咕是2011年中国颇具创新力的位置签到服务，也是基于真实地理位置和手机拍照的社交应用。

适用平台：iOS、Android、Symbian、Java以及MTK等众多手机操作系统

运营成绩：用户量300多万（手机客户端用户量），截至2011年4月，嘀咕签到总数达2000多万，日均登录用户2万人。

创始人简介： 李松，他是中国最大的手机位置签到服务商嘀咕网(Digu.com)、最大的相亲网站珍爱网（Zhenai.com）、以及领先的手机游戏服务商美思科技(Meuu.com)的创办人。在成为互联网创业人之前，李松曾任纽约投资银行贝尔斯登公司（Bear Stearns）债权部副总裁和摩根士丹利公司（Morgan Stanley）亚洲区执行董事。

产品特点：国内首创的MSD（Multi Sense Detector）技术，可以探测发现谁正在你旁边。结合了Foursquare与Color的成功元素，除了Color式的创新如探测和群拍，也借鉴了Foursquare的位置服务优势，引入本土化的用户主动签到机制（Check-in）。

营销心得：嘀咕网目前的产品推广大多是用户间的口碑传播，作为一种开放性的社交媒体，嘀咕用户多是通过位置关系、爱好关系、标签关系而聚集在一起的人群，相互之间进行信息沟通、信息传播的效率会更高。另外，通过用户的签到行为，将用户的足迹分享给其他SNS好友，这一裂变传播（One To N To N ）形式，也有助于产品口碑在用户间的相互扩散、相互影响和相互传播。

AdView

创立时间：2010年8月

核心产品名称：AdView移动广告优化工具

产品简介：AdView是专为Android、iOS开发者免费提供的移动广告优化工具，能帮助开发者轻松实现广告收益最大化。基于广告聚合的应用互推，能帮助应用迅速扩大用户数量。

适用平台：iOS、Android

运营成绩：用户量1400名开发者，应用数1500款，日广告点击10万；应用活跃用户数1200万/月。

创始人简介：林森，AdView团队项目负责人。天津大学计算机软件及理论硕士。曾任台资企业研发经理、微软中国开发经理，信产部和微软WINCE合作实验室主任。

研发感悟：影响产品的三个重要决策：决策一，移动广告优化产品必须保证公正的第三方角色；决策二，AdViw对开发者承诺永久免费；决策三，开发者的需求，是AdView不断完善的源动力。

北京千尺无限软件技术有限公司

创立时间：2008年

核心产品名称： 千尺下载

产品简介：一款手机高速下载的客户端产品，用户可以通过千尺下载更迅速、更便捷地获取无线互联网上的精彩内容，同时该款软件也是中国第一款拥有手机下载领域核心技术以及知识产权的产品。

适用平台：iOS、Android、Symbian、BlackBerry等

运营成绩：用户量3000万，日下载量超过1500万次，日活跃用户300多万。

创始人简介：冯文杰，2000年开始接触移动互联网，从发布WAP1.0的时候就开始做单机游戏开发，2001年参与创立数位红，2004年数位红被盛大收购后，2006年又参与创建3GV8。在冯文杰十年的创业经历中，绝大多数时间都是做手机游戏。后来看到迅雷在PC上的成功，通过分析手机软件的发展方向，决定放弃在手机游戏上的开发，而把未来的事业重点放在手机下载工具的开发上，于2008年创立了千尺无限。

产品特色：传统的下载工具可能是通过客户端的断点续传、多线程之类的方式把一些资源转到手机端。千尺下载更多的是运用云端的云加速技术，通过智能的分析用户所在的位置、网络等，然后缓存到不同的服务器中，这个技术被称作“千尺P2SP”，这个也是千尺独家在做的技术体系。通过对中国网络进行云端的加速，分析用户所在的省份、城市，用户用哪个运营商的网络，然后在用户的内网中进行匹配，理想情况下可以使用户的下载速度提升一倍。

重庆掌脉科技有限公司

创立时间：2010年9月

核心产品名称：掌脉智能手机跨平台阅读系统

产品简介：目前国内跨越最多3G智能手机操作系统的阅读系统。

适用平台：Android、WP7、iOS、Symbian（开发中）

运营成绩：用户量150万，下载量210万，活跃用户数140万。

创始人简介：陈旭，曾任电脑工程师，后加入媒体业，2010年7月开始着手创办重庆掌脉科技有限公司。

成功建议：1. 重视产品的商业策划；2. 微小的创新也是创新；3. 把用户体验放在首要位置，技术与商业均服务于此。

北京捷通华声语音技术有限公司

创立时间：2000年10月

核心产品名称：天行乐园

产品简介：天行乐园是一款集棋牌游戏、社区、论坛、聊天交友为一体的手机棋牌网游。其中在天行乐园android中创新性地融入了捷通华声语音合成技术。

适用平台：iOS、Android、Symbian、BlackBerry等

运营成绩：注册用户量300多万人，日登录量近20万次，活跃用户数近100万人

产品特色：内嵌自有专利语音合成核心技术等。

研发心得：努力将智能人机交互技术应用于移动产品中并悉心体会产品每个操作的感受。在原先语音合成技术中，人们总认为语音合成技术是死板没有表情的，甚至觉得它太无法接近了，与游戏这么娱乐性很强的产品无法挂钩，可是在经过了这样一次设计创新之后，发觉语音合成技术离我们并不遥远，它也可以化身为我们身边最亲近的“朋友”。

上海益实多电子商务有限公司

创立时间：2008年7月

核心产品名称：掌上1号店

产品简介：掌上1号店是B2C电子商务企业“1号店”（www.yihaodian.com）的移动客户端，提供给用户通过移动互联网平台所提供的从清单记录、搜索、购买、下单、到送货的一站式购物体验。

适用平台：iOS、Android、WAP、Windows Phone 7

目前运营成绩：目前掌上1号店尚处于试用期，以每日下载量不低于5000的速度持续增长。

研发心得：1. 了解用户的需求，以用户体验为一切的出发点；产品的创新和简化交互流程可以更好地提升产品的黏性。2. 企业要有自己的模式壁垒和产品特色。在运营方面，首先，要明确企业服务的用户群体的属性与移动电子商用户群体属性是否匹配，以及匹配程度是否满足企业目标。3. 根据现在移动互联网用户行为分析看，可开展的业务面还是非常广的，个人认为只要是具备实时性、便捷性、生活化的业务都会很有市场，贴近手机用户行为的业务服务也都会有很大的市场潜力。

国信证券股份有限公司

创立时间：1994年6月

核心产品名称: 金太阳手机证券

产品简介：金太阳手机证券是国信证券自主研发基于移动终端的综合移动证券服务平台，让投资者随时随地获取证券投资服务

适用平台：iOS、Android、Windows Mobile、Symbian、BlackBerry、J2ME

目前运营成绩：用户量362万，活跃用户数82万。

中国电信股份有限公司应用商店运营中心

创立时间：2010年3月

核心产品名称：天翼空间

产品简介：中国电信天翼空间是中国电信为广大用户提供的手机应用软件服务商城，提供给用户包括游戏、娱乐、实用工具、软件、书籍、音乐、旅行、社交网络等海量手机应用。

适用平台：Brew、Android、Windows Mobile、Windows CE、BlackBerry、Palm

运营成绩：截至2011年6月10日，注册用户数2300万，活跃用户数超500万，应用上架数2万，下载量2800万。

北京沃安科技有限公司

创立时间：2009年11月

核心产品名称：“炫拍”——手机视频分享

产品简介：最便捷的手机视频分享平台，为用户与手机通讯录好友、站内好友、微博社区等提供跨平台、端到端、实时/非实时视频分享服务。

适用平台：iOS、Android、Symbian

运营成绩：炫拍服务从2011年5月6日试运营（开放测试）以来，截止5月底，用户量2600，下载量4100，活跃用户数600。

创始人简介：曹学武，沃安科技创始人。曾经在中兴通讯、量子无线、宇讯科技、RCG China 等公司工作。

成功建议：考虑用户需求和市场的发展方向，寻找适合自己的市场机会和产品定位。扬长避短，集中精力把自己的业务领域做精。

安徽科大讯飞信息科技股份有限公司

创立时间：1999年6月

核心产品名称：讯飞语音输入法

产品简介：讯飞语音输入法集语音、手写、拼音输入于一体，是全球首款基于云计算方式实现的智能语音输入法。

适用平台：Android

运营成绩：用户量大于100万，下载量大于500万，活跃用户数大于40万，用户量大于100万。

创始人简介：刘庆峰，1990年考入中国科学技术大学，先后获通信与电子系统专业硕士学位与信号与信息处理专业博士学位。中国科学技术大学兼职教授、博士生导师，中华全国青年联合会委员、中国科协七届委员和十届、十一届全国人大代表。1999年创办安徽科大讯飞信息科技股份有限公司。

成功建议：比用户自己更懂用户，小步快走，迭代优化，细节决定成败。

金蝶理财网

创立时间：2010年3月

核心产品名称：金蝶随手记

产品简介：记账就要随手记。

适用平台：iOS、Android

目前运营成绩：系列产品总下载超过 1000万，收费版本用户超过20万，已连续10个月占据App Store中国区财务分类第一。

创始人简介：谷风，2000年加入金蝶集团，历任金蝶软件互联网事业部总经理、金蝶友商网副总经理、金蝶随手网总经理。从2004年开始负责金蝶在移动互联网方向的探索，创建了多个金蝶内部创业团队。2010年3月，带领团队开始开发随手记 。

成功建议：1. 前提是“了解你的客户并关怀他”。2. 一个产品解决一个用户问题，别臃肿，清晰明了。3. 由于应用商店的排行竞争，移动产品比互联网产品还要快，控制版本周期，快速迭代。4. 商业模式上思考清楚，移动互联网机会窗时间有限，不要纠缠在必败的战斗中，迅速选择。

北京推图科技有限公司

创立时间：2011年3月

核心产品名称：推图

产品简介：一款时尚有趣的手机拍照分享应用，随时用照片和朋友分享生活。

适用平台：iOS、Android

运营成绩：用户量11万，下载量15万，活跃用户数5万

创始人简介：陈仲军，1998年四川大学通信工程专业毕业，13年互联网从业经历，曾在太平洋电脑网任开发部经理，在惠聚网担任技术总监职位专门从事移动互联网产品开发；较早涉足移动互联网，领导开发了多款移动互联网应用。

成功建议：潜在用户群体一定要够大，选择正确的时机、正确的核心种子用户，把握好开放平台的大好时机，做好运营。尤其强调的是要做好“社区营销”，特别是事件营销，以及用户间的口碑营销。

珠海市伟衡科技有限公司

创立时间：2011年1月

核心产品名称：圈圈

产品简介：协助手机用户创建、维护、分享和拓展在不同时间地点形成的特定人际圈。

适用平台：iOS、Android

运营成绩：尚处于邀请测试阶段。

创始人简介：吴志力，1998年考入清华大学，2002、2004及2008年分获香港浸会大学计算机学士、硕士及博士学位，研究方向为机器学习及数据挖掘。曾于2005年任美国密西根大学统计系访问学者。2009年获Google研究奖（Google Research Award），并被资助于英国利兹大学从事博士后研究。2011年创立伟衡科技，率团队开发的“微博分析家”荣获2010年春晖杯创业大赛一等奖，并率先开发出乞讨儿童数据库、公益随手拍等社会反响良好的手机公益应用。

成功建议：设计主导，产品的用户体验、用户界面与功能深度结合，需要主导产品的发展。高效编码、拥抱新技术，比如HTML5有利于产品原型快速开发。国际视野，需要从国际市场来观摩借鉴最新的创意。推广门路和渠道很重要，但要靠好产品说话。

从数据看 iOS移动应用开发

口述 / 熊俊　　整理 / 长生

iOS 移动应用开发和营销存在许多值得推敲的细节和成功诀窍，本文根据厦门同步网络CEO熊俊在2011年4月28日由长城会和CSDN联合举办的“开发者星球TUP”大会上的演讲和会后采访整理而成。

iOS方面有哪些值得开发者们关注的内容和数据？

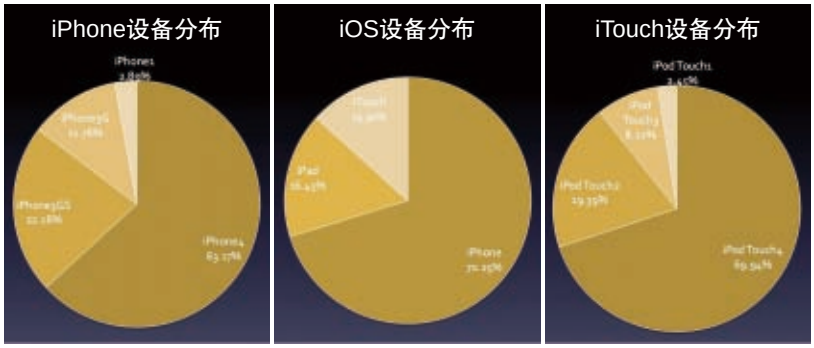
今年三月份时，苹果公布了iTunes Store的数据，包括音乐、游戏、软件，现在一共有两亿多个注册账号。相对于苹果的iOS，现在全行业的人都认为Android是非常好的系统，我也认为Android的发展速度非常快，但关于它的数据大部分来自于华为、中兴等低端手机，大量平台割裂和分散的推广渠道给数据分析带来很大的困难。但是1.5亿iOS设备所对应的2亿账户，数据对应清晰利于分析。

iOS以及移动设备分布概况

中国究竟有多少使用iOS的移动设备？这是大家都想了解的数据，根据我们现有的数据再综合一些设备生产厂商的数据，可推测出全球有将近1.5亿，其中中国境内现有1千万使用iOS的移动设备，北美大约有6~7千万。使用iOS的移动设备包括iPhone、iPad、iPod Touch，我们发现国内使用iOS的移动设备排第一名的是iPhone，其比例大约占总数的70%。国内的一些朋友刚开始都觉得iPhone很贵，但买了后，iPhone就不是一个纯粹意义上的智能手机了，在很大程度上是一个炫耀的工具。我们可以看到现在地铁里有很多人拿着iPhone 4，感觉它已经变成街机了。在北美，iPod Touch是小孩子最想要的礼物，而在国内买得起iPod Touch的消费者稍微咬牙就可以买一个iPad。国内的1千万使用iOS的移动设备中，iPhone有大约700万用户，而这其中又有63%是iPhone 4的用户。可见iPhone 4是国内使用最多的iOS移动设备。

而iPod Touch只占13%，13万用户中又以iPod Touch 4的用户数量居多，有将近70%，iPod Touch和iPhone的版本比例很相似。

图1 iPhone、iOS、iTouch 设备分布概况



App Store上的应用现状

在App Store上的应用包括书籍、健康、商业、教育、娱乐、游戏等20类，在分类应用所占比例中游戏的份额最高占总体的16%，其次是书籍占15%。书籍应用的比例很高，是因为每本书都可以做成一个应用。制作过程非常容易，主要是做一个阅读器，然后找一些有版权或没版权的书籍往上一放，就成为了一个书籍应用。苹果应用商店里有大量的书籍应用，这牵扯到版权问题，我们看到中国的很多作家协会开始将维权的目标指向苹果应用商店。从分类应用所占比例中可以看出，开发者都认为游

图2 App Store 各分类应用占比



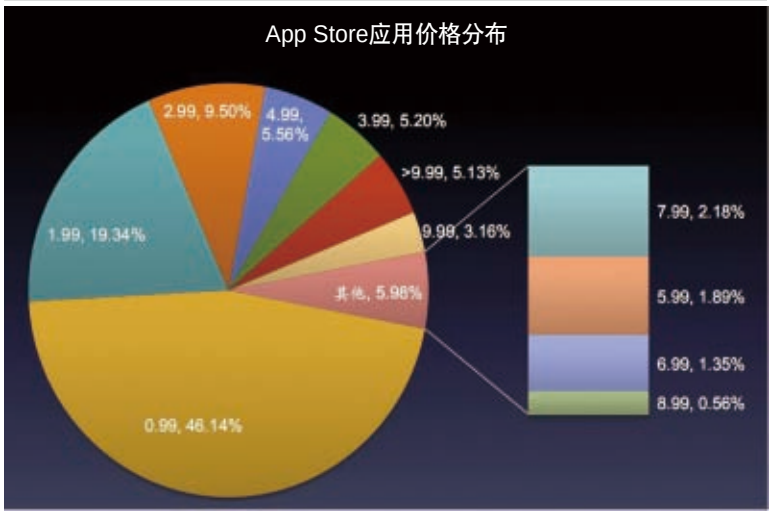
戏是很容易赚钱的应用，所占比例较大的还有娱乐、教育、生活和工具，一些专业类的应用相对较少。

App Store上的应用收费情况如何？从已知的数据中我们可以看出在应用商店里有63%的应用都是要收费的，免费应用只有不到37%。在收费应用中有4%的应用收费金额在0.99美元以下，也就是6~7元就能买到这个应用。随着价格的增加，应用的销量逐渐减少。有些应用的价格高于9.99美元，估计开发者的潜台词是既然应用总是卖不出去那就卖贵一点。苹果应用商店的软件排行榜分为收费、免费和销量排行。销量排行榜中的名次排位不光和下载量有

图3 App Store 应用收费情况



图4 App Store 应用价格分布



关，还同销售额有关。根据苹果应用排行榜之前的规则，收费应用的销售额达到三万美元就基本上可以排到第一名。因此我们可以在销量排行榜中发现经常会有价格为999美元的应用排在前列，而这些应用的销售量其实并不高。

截止2011年4月20日左右，在App Store里共有384491个应用，从新品上架、员工最爱、收费、免费和销售等排行上能被展示的应用不会超过5千个。另外，有38万款应用是用户很难通过浏览的方式发现的。在中国区如果你的应用能在免费排行榜上待一天，会得到十万甚至十几万的下载量，而如果你的应用没有上榜，下载量就只有区区几十。这可以类比到Android应用上面，让用户看到应用，可提升应用下载率，所以在未来，开发者面临的一个大问题是如何增加应用被展示的机会。

如何提高自己的应用销量

这里，我介绍一下开发者们为提高应用的曝光率和销量，需要重视的几个方面。

第一，**命名**。软件应用的命名其实和网站的SEO是一样的，要选择容易被用户搜到的名字，例如热门应用。如图6所示，这是在同步助手里的搜索热词排行，统计了前20为搜索量的百分比。可以看出QQ这个词的的搜索率非常高。单纯看前20名热门词汇的搜索量比例，QQ与HD之间相差将近三倍，所以中国用户对QQ

图5　iPhone&iTouch 应用装机量排行

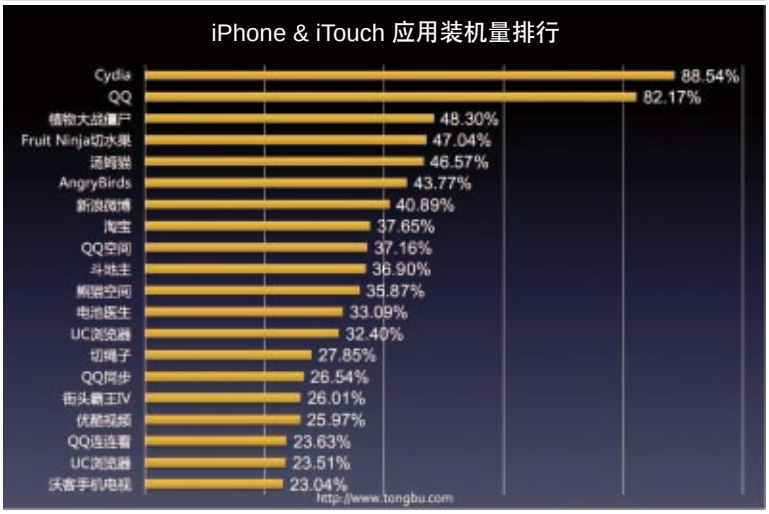
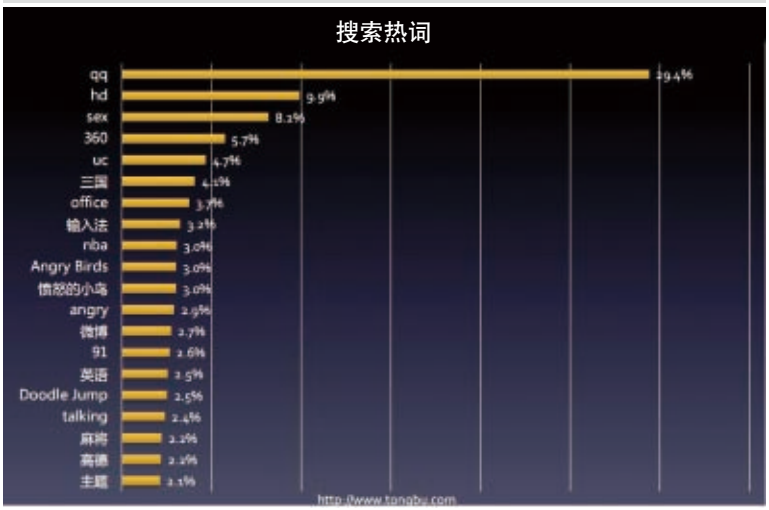


图6　同步助手搜索热词



的依赖还是非常高的。360软件虽然没有上苹果应用商店，但是我们知道现在搜索360会出现一堆和360有关系的软件，如果开发者做一个应用，名称里有360这个词，被搜索出来的几率就增大了。如果你把搜索热词中的多个关键词组合起来请名字，应用被搜出的概率就更高了。

第二，**应用大小**。如果是工具类，一定要清楚它对用户来讲要有功能上的期望，既不能太大，也不能太小，一般在1~3M之内。在Symbian时代，大家都用WAP，搜索慢、流量有限，应用太大会对用户的下载有心里障碍，但是现在都是iPhone了，3G、WiFi已经没有流量上的限制，所以大小正成为用户考量应用价值的标准。大家都有一个直观的感受，软件越大里面放的东西越多，如果软件小，大家会觉得，软件为什么这么小？是不是很差？如果软件名字很好、图标精美，软件容量还比较小，用户可能觉得还不错，但图标做得差、很山寨，又很小的话，用户会直观感觉是一个很差的开发者做的，品质不会好到哪里去。如果是游戏类，适合集中在几十M到100M之间。如果大于500~600M对于用户来讲也有下载的压力。

第三，**应用下载时间**。应用上线后在什么时间推广比较好？iPhone用户的活跃时间是从北京时间8~9点开始，这时来自中国的下载量会支撑起第一个小高潮，真正的高峰是在10~11点，这时大家往往有时间在手机上下载应用。所以应用上线的时间就一定要在高峰期之前。

第四，**用户网络**。使用iPhone的用户中多数还在用移动网络有57.9%，用中国联通网络的有39.4%，这说明使用移动网络的人还是非常多的。其他占2.7%，这些都是用国外运营商电信卡的人，包括一些美版手机用户使用的卡贴。所以我们要思考到如果做一个面向国内用户的应用，就要考虑到国内用户中还有很大比例在使用中国移动的网络，也就是2G网络，速度很慢，网络体验也不好。再加上2.7%的其他运营商用户，一共有六成用户的网络状况不佳，而联通3G用户存在信号问题，所以在做应用时要充分考虑用户网络的相关影响因素。

第五，**多途径营销**。对开发者来讲，在苹果软件应用商店曝光应用的机会是非常稀缺的，所以一定要充分利用一些其它合法的渠道，比如说一些免费的应用，可以找威锋或iapp、同步助手、91来做推广。对于收费的应用可以灵活一些，除了增加曝光几率，还可以充分利用苹果应用商店的兑换码、推广码，比如多给一些媒体做评测，以争取最大的宣传效益。



熊俊

同步网络创始人，CEO，91手机助手作者、创始人；2010年离开网龙公司无线事业部后，创办厦门同步网络有限公司。

雅虎重拳出击移动数字传媒

——雅虎首席产品官布莱克·欧文专访

记者 / 长生

作为全球用户最多的互联网公司之一，雅虎近年来一直致力于转型，以便在移动互联网新时代里取得新的辉煌。去年就任雅虎全球首席产品官的布莱克·欧文（Blake Irving），无疑是雅虎振兴计划中的核心人物。欧文曾经在微软从事了15年的产品研发，被业界誉为“MSN Messenger之父”，如今执掌雅虎的产品线，自然会有与众不同的考量。而新近推出的Yahoo!Livestand，可谓他在移动互联网领域的一次重拳出击。

Yahoo!Livestand是一种针对平板电脑、智能手机设计的数字报亭，将允许出版商和广告商在上面无缝发布内容，并根据消费者的兴趣提供个性化的阅读体验。近期，在北京举行的Yahoo!Livestand产品演示会上，《程序员》记者有幸采访了布莱克·欧文。

记者：作为Yahoo!的首席产品官，能否介绍一下Yahoo!产品体系的整体战略愿景？

布莱克·欧文：我们有以下五个战略要素。

第一，始终以用户作为我们服务的中心。

第二，向用户提供跨平台的服务。

第三，建设基于科学服务的平台。比如对于注册用户，进入网上商城后，可以享受到个性化的商品推荐服务。

第四，建设一个社交关系网络，内涵是用户可以和朋友们共同分享一切感兴趣的事物。

第五，打造一个由广告商和出版商共同组成的产品开发系统。

记者：新近推出的Yahoo!Livestand将为消费者、出版商和广告商提供哪些优势？

布莱克·欧文：平台电脑和移动电话的使用正在爆炸性增长，但数字媒体却没有跟上。消费者无法找到在报摊上买到的出版物，出



布莱克·欧文认为，Yahoo！社交网是以兴趣为基础的

版商和广告商无法触及想要服务的受众。而Livestand是个让人沉浸其中的环境，为消费者提供动态和个性化的体验，为出版商和广告商提供新鲜和主动内容的传送途径。它有如下优势：以个性化的内容绕过网页的垃圾信息；允许出版商向受众大规模传达个性化的内容；还可以为广告商重新规划数字广告。

记者：你提到了社交互动，与Facebook相比，Yahoo！的社交网有哪些特色？

布莱克·欧文：Yahoo！社交网是以兴趣为基础的。兴趣网和传统社交网完全不一样，比如在兴趣网中，我对自己的兴趣描述，会比三年前的更有针对性、更真实。另外，随着时间的推移，社交网可能会变得越来越大，于是朋友之间联系的密切程度会被冲淡，但兴趣网完全相反，对兴趣的反映能越来越精准。

记者： Livestand目前的盈利模式是什么？

布莱克·欧文：实际上Livestand应用就像Android应用，对于普通用户本身是免费的。我们会对出版商收费，但Yahoo！会把出版商的广告进行系统性整合，放到Livestand平台，使得出版商也可以通过Livestand赚钱。这种广告和出版的交换，实际上与搜索引擎的商业模式是很相似的。

与平台共舞，Opera的成功之道
——专访Opera中国研发团队

记者 / 杨东杰

1992年，挪威最大的电信公司Televerket（后更名 Telenor）的两名工程师在操作电子计算机的时候，对当时市面上唯一的一款商用浏览器Mosaic感到不满，觉得它体积庞大而且功能单一，就决定自己开发一款小巧、好用的浏览器，这便是后来大名鼎鼎的Opera浏览器的雏形。从1995年Opera软件公司正式成立至今，在浏览器这个平台必争之地，Opera独辟蹊径，以创新和品质为核心竞争力，成功发展成五大浏览器品牌之一，也是目前唯一的一个成功的跨平台第三方浏览器厂商。近日，《程序员》记者来到Opera中国，专访Opera中国工程总监黄奔阳和公关总监高伟，了解Opera与平台级厂商竞争并成功的发展之道。

成功经验：跨平台战略、专注和创新

进入移动互联网时代，人们才发现，在PC之外，Opera已成为数一数二的浏览器品牌。



Opera中国公关总监高伟指出，Opera的定位是跨平台浏览器

Opera中国市场总监高伟告诉记者“Opera对自己的定位是跨平台的互联网浏览器，事实上，Opera在包括 Mac、PC 和 Linux 计算机、手机、PDA、游戏终端以及其它联网设备如电子书、智能家电上都有极高的覆盖率。”

Opera是现存历史最悠久的浏览器品牌，和它同代的是Netscape和IE，正是因为小巧轻量的特性，在Opera面向IE近乎毁灭性的平台捆绑竞争时，可以快速将市场切换到PC之外的广大嵌入式设备，避免了同Netscape一样的陨落命运，在后来的浏览器大战中，保持了独特的竞争优势。

尽管产品遍布多个平台，而对于Opera公司来说，只是专注于生产一款产品——Opera浏览器。因为所有的浏览器都是使用同一个内核，只是根据不同平台的特性对功能和UI进行封装。

正是这种持续的关注打造了Opera高品质的口碑，全球主要的平台商都和Opera建立了良好的预装合作。以Opera中国和MTK的预装合作项目为例，测试团队持续测试四个月，Opera浏览器的测试结果是零bug，完全达到MTK的工业级品质要求。

专注也使得Opera能够不断引领浏览器创新，标签式浏览、页面缩放、鼠标手势、内置搜索、快速拨号等众多重要功能创新都来自Opera，后被业界普遍采用。

融合骇客精神和工程师文化

令人惊讶的是Opera基本不申请专利，即使是“内嵌搜索”这样成为浏览器主要盈利模式的创新。“一方面是因为Opera作为技术主导型公司，有着浓厚的工程师文化和骇客精神，另一方面Opera对自己的研发团队非常自信，相信他们



Opera中国工程总监黄奔阳认为，Opera中国的优势在于平台技术和创新力

一定会不断创新，保持领先。”高伟解释道。

Opera如何保证行之有效的创新机制？据介绍，Opera提供了如下四方面的措施。

骇客群体。目前，Opera全球大概有近800名员工，其中600多名是研发人员。Opera的人才理念是“只招聘聪明人（Opera 创始人谭咏文语）”，所以有很多骇客式的员工，他们对技术痴迷，乐于分享技术心得和创新想法。

彻底的分享。Opera任何一位开发人员都可以无限制地接触到公司所有产品的源代码。在Opera的内部邮件列表上，不断可以看到各种产品点子和技术特性的讨论。

创新融入日常工作。在Opera并没有“创新中心”这个设置，只有工程中心，Opera也不像Google一样有规定可以用20%的时间专门做创新。因为Opera认为，创新来自日常工作，并不需要刻意划分。Opera许多重要产品例如Opera Mini都来自工程师在工作中迸发的灵感。

打破一切界限。尽管Opera以产品研发为主，但公司的其他部门都可以自由参与到产品研发中，包括销售和市场人员。

中国研发团队：最具创新力

Opera目前有4个工程中心，分工各有侧重。其中位于挪威奥斯陆的工程中心主要是负责Opera浏览器内核的开发；瑞典的工程中心则主要负责Opera Mini和跨平台UI；波兰的工程中心更多地承接了客户支持的工作。Opera中国也是四大工程中心之一。

Opera中国工程总监黄奔阳告诉记者，Opera中国的工程师团队并不做汉化之类的本地化工作，Opera的汉化是外包给一家翻译公司完成的。Opera中国的优势在于平台技术和创新力。Opera中国有很多平台专家，承接了很多Opera拓展新平台的研发工作，例如Limo、Android、MTK和Spreadtrum等版本。

但Opera中国最受总部认可的是突出的创新能力，Opera中国提交的项目总能够在总部行动之前，就命中重要的发展方向。一个典型的例子就是Opera Turbo项目，Opera Turbo项目的目的是将原先用于Opera Mini的网页压缩技术集成到Mobile，它推出的意义不仅在于提供用户更好的浏览体验，对于Opera来说，还在于实验Opera Mobile和Opera Mini的融合。

Opera Mobile和Opera Mini都是面向移动终端的浏览器产品，区别在于Opera Mini 是瘦客户端浏览器，通过服务器压缩网页，最高可达90%，可以加速浏览并节省资费。Opera Mobile则是一款全功能的手机浏览器，将页面下载到本机进行渲染，类似于电脑上的 Opera 浏览器。这两个产品在Opera产品系列当中层级相同，对于用户来说，只关心浏览体验，并不在乎版本的差异。目前这两个产品的团队已经开始合并，未来Opera Mobile和Opera Mini极有可能融合成一个产品。

基于Opera独特的研发管理体系，每个工程中心所承担的任务很大程度上取决于这个团队的能力。而中国团队则被认为是“最具战斗力和创新力的一支”，因此承接了大量重要项目。但由于Opera对人才要求极高，从2006年Opera进入中国至今，团队成员才扩张到50多人，工程总监黄奔阳不得不放缓发展速度以保障精英团队发展路线。

今年，Opera在中国最重要的动作是三月份与天音通信共同组建合资公司。Opera作为一家技术型的国际公司，在国内的推广难免有些水土不服。合资公司将专注于本地化业务，推出适合中国人使用习惯的产品和服务。”

Android市场将持续爆发

——点心OS软件架构师梁泉专访

记者 / 许坤铭

近期关于Android 发展遭遇瓶颈的话题愈演愈烈，业界对于Android现状的关注度也日益提升，为了帮助开发者们解答Android发展前景的问题，《程序员》杂志对点心OS的软件架构师梁泉先生进行了采访，以了解国内行业专家对于Android 发展趋势的最新观点。

记者：在Android发展历程，哪些因素是成功的关键？

梁泉：有几个重要的因素。

第一，**开放**。这是最关键因素。谷歌对待Android的开放程度是前所未有的，商业上的免费模式，打破了以往平台的授权方式，降低厂商成本的同时也提升了他们参与的热情。技术上的开源策略，让开发者可以参与改进优化，以最快的速度催熟平台。如今绝大多数移动设备芯片商和制造商，都已将Android当作发力智能设备的最佳平台。

第二，**技术选型**。首先是采用Linux作为底层操作系统，Linux在嵌入式上的沉淀让Android得以方便地移植到各个平台并快速发展。其次是通过虚拟机，支持使用Java开发应用框架和应用程序，极大地降低了Android开发者的门槛。最后是Linux和Java的选型，都非常契合Android开源的策略，两种技术在开源社区都有着丰富的资源，Android的成功，既可以算是开源社区的结晶，也让历史悠久的开源社区有了新的寄托。

第三，**时机**。Android推出的时间点，正是第一代智能手机操作系统略显疲态的时候。Android一开始虽在用户界面和体验方面乏善可陈，但逐步也跟上iOS的步伐，相反竞争对手因为固有的设计缺陷，开始快速下滑。硬件的发展也给予Android足够的配合，Android已经迈过早期性能障碍导致的发展缓慢时期，进入爆发期。

其他因素还有两次Android程序设计大赛，带来最早一批优良应用的同时，也推动建立起最基本的应用商店。另外，一开始选择高通成熟的智能手机芯片方案，并选择已有丰富智能机经验的HTC作为最早合作伙伴，也极大地促进了Android的成功。

记者：您觉得哪些地方是Android系统做得不够的？

梁泉：Android目前有几个问题。

首先，**分裂**。这是开放必然带来的问题，一度非常严重。各种性能CPU，集成GPU与否，各种屏幕分辨率，电阻/电容屏差别，都给开发者带来重重障碍。所幸谷歌早已认识到此问题，通过控制新版本发布节奏等方法进行调整，效果显著。数据显示2.1跟2.2版本占据市场绝大部分份额，接下来2.3的比例在逐步提升，加上硬件规格越来越高，同时开发适配非主流的屏幕周期较长，Android设备硬件规格也渐渐拉近差距。点心OS作为Android的增强版本，一直以来从不提倡分裂，在提供系统改进及增强功能同时，从整体上做到与Android完全兼容。



点心OS架构师梁泉认为Android的首要问题是“分裂”

诸如保证完全不修改原有API接口，保证完全通过CTS等，都是点心OS一贯坚持的策略。

其次，**安全**。Android的安全风险关注度一直较高。除了破解、刷机带来严重的安全问题外，Root权限轻易被应用获取，使得系统权限机制形同虚设，近期还有窃取用户隐私（如联系人、短信等）的恶意程序出现。暂时还没有看到谷歌的动作，因此可以预见在未来很长时间里，Android用户都会面临此类安全问题。

此外还有不少技术问题，试举几例，如Skia缺乏硬件加速导致GUI绘图低效、被应用程序滥用的wakelock机制导致严重耗电问题等。

记者：有哪些竞争对手将会成为Android的劲敌？

梁泉：这个问题要从不同角度来看。

从将来设备和用户数量方面，我认为目前已没有竞争者是Android的对手。 Android在产业链上已形成足够大的生态系统，和苹果封闭的产业链策略相比，支持Android的芯片商、设备制造商及运营商所形成的合力，远远超出苹果的能力。Android芯片商远不止目前的高通、TI、三星及nVidia，设备制造商也远不止出货量大的HTC、三星、摩托罗拉及华为等，未来两年，Android平台设备总量会远超过iOS平台。

但从设备的体验、应用程序/游戏的数量及质量等方面，这个问题可能会有不同的答案。苹果软硬件一体化设计带来了良好的用户体验，而开发者的体验也是一致的，开发高效的同时回报也远高于Android平台。因此相对于Android，iOS依然是很多开发者的首选平台，很多优秀应用/游戏，也都在iOS平台首发。Android和iOS都具备成型且成功的应用生态系统，但目前iOS依然略胜一筹。

至于其他竞争对手，市场规模可能很难做大。MeeGo有根本的问题， Intel x86耗费大把精力在功耗等问题上，基于Qt的软件开发门槛也较Android平台的Java框架更高，同时MeeGo推出时间已经晚了太多。Windows Phone 7情况会稍好一些，但依然缺乏应用开发者资源，开发者很难同时投入多个平台，已有足够市场份额的iOS和Android会牢牢拽住他们，生态系统很难有效建立起来。如RIM或HP乃至三星，都有自己的操作系统平台，尽管这类平台依然会有一定用户支持，但暂未看到能追赶Android的迹象。

记者：和竞争对手们相比，谷歌Android的优势和劣势都有哪些？

梁泉：抓住先机，采用开放策略所营造的生态系统，已是Android最大的优势。无论设备还是用户数量，Android的未来都有良好的预期。而劣势除了上一问题已经分析的因素外，还有和iOS这样的软硬件一体化平台之间的差距。

记者：关于谷歌Android发展遭遇瓶颈的说法，您有什么观点？

梁泉：我认为这是非常片面的说法。我也看过相关报道，其仅仅源自对Android应用开发商的一些调查，以及跟iOS的一些对比分析，而非从产业链的角度来看待Android的发展问题。前面已经提到，随着支持Android的芯片商和设备商越来越多，Android目前呈现爆炸性增长的趋势。有更多的设备，有更多的用户，应用开发商即使因为分裂问题等有所犹豫，但很快他们又会回到Android。

记者：在谷歌Android未来的发展中，有哪些关键的因素是需要注意的？

梁泉：最关键的问题，仍是对待开放策略的态度。开放造就Android今天的成功，也带来分裂、安全一系列问题。这里的开放，不是简单的问题，在产业链如此细分的今天，苹果可能是逆势而为方面绝无仅有的成功者，但谷歌绝无可能走向封闭而自断生路。不过它确实需要在接下来的策略里如履薄冰，平衡各方利益，让整个生态系统积极发展。苹果最近发布会上出现的iMessage等应用，引起很大争议，甚至被认为是“抄袭”，这会极大打击开发者积极性。相比之下，谷歌生态系统里角色更多，也更复杂，3.0暂时封闭就让很多人紧张，因此谷歌更需要谨慎地考虑每一步决策。

另外如安全问题，谷歌正着手处理，但同时这也需要开发者，包括像点心OS这样的系统开发者，都参与进来，一起以开放的心态推动改善，将潜在风险扼杀在摇篮里。若现在不及时处理，将来这类问题爆发，有可能对Android造成毁灭性打击。毕竟用户至上，如果用户对平台失去信任，再好的形势也可能急转直下。

手机游戏：崇尚简单至上和零碎娱乐的永恒价值

文 / 郑金条

相信用户的选择

对于手机游戏，在创造用户需求这个诉求太缥缈的时候，也许相信用户的选择是更好的研发趋势。

Information Solutions Group曾对玩家选择手机游戏的倾向进行过追踪，所得到的结果并不让人诧异，该调研称玩家会分享的选项包括可玩性强、容易上手、多重关卡等，有持续游戏性、有挑战性、有动人心扉的声效、能够消遣解闷的同时，具有一些不同以往的独特玩法和特征。

大量的开发者深谙其道，Rovio（代表作Angry Birds）公司 Ville Heijari认为简单娱乐是游戏的首要诉求（Simplicity was always the top priority），而PopCap（代表作Plants vs. Zombies）公司Andrew Stein认为让玩家易于上手至关重要（Be quick to play）。



一些被实践的美好

尽管现在手机游戏领域已经进入了以活跃用户为评估指标的时代，但通过下载量，我们依然可以赞叹这些顺势而为的公司们的超级成功。

Rovio（代表游戏Angry Birds）实现了超过2亿次的下载；Backflip Studios（代表游戏 Paper Toss）实现了超过1亿次下载；Outfit7（代表游戏 Talking Tom Cat）实现了超过1亿次的下载；Chillingo（代表游戏Cut The Rope）在苹果App Store实现了1.4亿次下载。



如果从部分手机应用商城的下载比较就可以更鲜明地看到这些崇尚简单至上的公司所获得的让人惊诧的下载量：Samsung Apps从2010年6月到现在也才下载1亿次，中国移动应用商城在2010年底也仅实现了1亿次的下载。

面对似是而非的质疑

尽管类似Rovio依托Angry Birds这样简单的游戏迅速崛起甚至成为市场和资本角逐的对象，但是相对于掌机游戏和视频游戏而言，它们在画面（相对粗糙）、操作方式（简单）和音效（边缘）各个层面仍然是各方质疑的对象。

华尔街日报曾经有刊文详细解析了这个问题：那些看似聪明的人究竟为什么能够沉迷于这些简单的小游戏？该篇文章对这个问题的终极解答是这类闲趣游戏是人类日常生活中可以随处存在的“娱乐小点心”，以可爱温馨的画面、逗趣的声效以及取悦玩家的奖励系统来娱乐玩家的零碎时间和释放他们的生活压力。



事实上，这些简单的游戏有些甚至充满了艺术造诣。比如Angry Birds曾获得GSMA的最佳游戏大奖；Cut The Rope获得过BAFTA和GDC的最佳掌上游戏大奖以及WWDC的最佳iPhone应用设计大奖；Infinity Blade获得过IMGA和WWDC的最佳游戏设计大奖。这些奖项并非可有可无的边缘奖项，而是当前最受行业肯定的专业评审，所以崇尚简单也同样能够拥有娱乐的无限价值。

简单游戏的体面成就

前文提到的Infinity Blade按照Epic Games的对外说法，该游戏在苹果App Store发行半年的时间内就获得了超过1000万美元的净营收（单款售价5.99美元，其中43.7%的营收来自内置付费功能）。这是一款画面精美但是以展示虚幻3D引擎为主的简单玩法游戏，从头到尾饱受用户的诟病，但是也正因为其简单玩法带来的对战乐趣让这款游戏持续停留在排行榜中。

不仅如此，苹果App Store曾出现过一个相对有趣的局面，来自韩国Gamevil的Air Penguin、来自芬兰Rovio的Angry Birds Rio、Angry Birds和来自德国Andreas Illiger的Tiny Wings这四款以鸟类为主题的简单游戏同时出现在了排行榜的前四位，它们的共同特点在于满足了用户对于简单快乐的需求。Gamevil美国总裁Kyu C Lee讲解Air Penguin的成功时认为游戏的简单玩法和带来的无穷乐趣是游戏能够在短时间内获得用户认可的关键环节。

如果要仔细计较的话，看苹果App Store全球各个国家和地区的详细分榜单可以很明晰地看到简单至上的Air Penguin、Angry Birds、Tiny Wings、Fruit Ninja、Words With Friends、Cut the Rope、Doodle Jump和Plants vs. Zombies是各大榜单中最有下载竞争力的游戏。

简单的价值来之于创意性

简单闲趣的游戏其核心价值在于创意玩法。

Anthony O'Dempsey的One Single Life以自由狂奔者的一次性命运给了玩家非此即彼的选择和对死亡的谨慎以及震撼，这款游戏渗透了对玩家心理层面的追踪和研究，尽管没有盈利模式和诉求，但对开发者特立独行的设计架构方式很容易形成品牌印象。

这种玩法的独特性在于不仅有趣而且不落窠臼，类似Chillingo发行的Spider Jack因为和Cut The Rope相当接近，但即使从宣传上吹得天花乱坠也很难从挑剔的玩家那里获得一致性的赞誉口碑。

创意是什么？是不是就像Mathieu Nouzareth所说的“单纯创意可能只是一颗华丽丽的鸭蛋”？

创意附属：纯创意、综合借鉴和纯山寨的辨析

游戏邦曾发起过一次关于创意价值和山寨得失的讨论，其实这个领域充满了价值博弈，也不乏从山寨模式成功起家的公司。

但更真实的一面是山寨只能是极少部分可能成功，更多的开发者都在山寨中沉沦。如果没有赶上好时代（平台开放前期+不同平台的空档差），不支持创意已经很难从资本和资源型市场脱颖而出。尽管我们偶尔还能看到不少产品在苹果App Store、Google Android Markret甚至国内和国外环境的差异化以及发布空档中抢到了生存的空间，并可能还因为这些空间的存在而表现出了暂时的强势价值。

山寨看起来是一个更安全的生存方式，App是个竞争激烈的环境，而山寨被证明成功过的游戏至少在题材和玩法上更可能贴近玩家的需求，从而降低遭遇市场抵制的风险。但事实上，这是对从业环境的顾虑，包括如何在竞争激烈的环境下首先获得生存的考量，只是这种既定的削足适履的仿制行为将可能带来更多的后遗症，包括：首先是品牌污点，任何公司都有可能在市场的浪淘沙中崛起而成为潜在的超级公司，而山寨起家的污点对之后品牌声誉的影响将如影随形，再多的公关工作也很难弥补山寨的负面效应（Fluik旗下的Office Jerk明显具有Backflip Studios旗下Paper Toss的风骨、Dillo Hills基本和Tiny Wings有相似的模式而

Spider Jack在游戏玩法上和Cut The Rope则几乎如出一辙，即使Office Jerk宣扬得轰轰烈烈，但它的价值始终笼罩在Paper Toss阴影之下）；其次，可能遭遇诉讼，这种诉讼肯定将之前可能不会爆发的品牌污点瞬间揭示在公众的眼皮底下，假使败诉将对企业未来的持续发展带来灭顶之灾；此外，可能还会造成企业内部的创造力惰性，任何事情都只想着从借鉴和山寨开始，很难形成自己独特的品牌和文化价值（包括独特价值来驱动消费力），甚至会伤害企业成员的积极性，一个没有创造力的环境很难留住满怀抱负的顶尖人才。

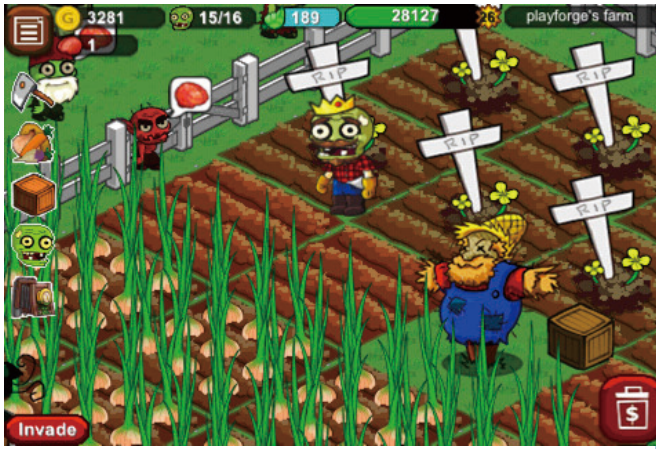
当然假设一个企业能够靠山寨最终成为行业的领先者，那么它将遭遇到曾经被山寨方的困局，它可能也面临着被山寨，并且可能没有足够的话语权从知识产权和法律的角度抵制山寨的二次冲击。

其实，我们真正想说的是游戏也是一种文化行为，每个人都能够在感受玩家需求的基础上做出吻合市场价值的产品。开发者的每款作品都在传述他们自己对引导用户娱乐方面的独特见解，这种见解将以画面、机制甚至声效等各个环节渗透到用户的时间成本中。

事实上所有人都应该更有见地地相信，在移动互联网时代信息的交流没有任何阻隔，每一个玩家都是精明的玩家，他们对游戏题材和玩法新鲜感的追求远远超越我们的想象，纯粹山寨而期待平台差异和发布时间的空档差异并最终奢望能够获得成功的可能性将越来越渺茫。

这些在苹果App Store排行榜长盛不衰的简单游戏（Air Penguin、Angry Birds、Tiny Wings、Fruit Ninja、Words With Friends、Cut the Rope、Doodle Jump和Plants vs. Zombies）依托的并非山寨的理念，而是从玩家的角度设想，试图阐释他们对零碎化娱乐的理解。Angry Birds只需要玩家敲定下抛物线的简单投放路线，而Tiny Wings更是只需要把控这只短翅膀的小鸟起飞的时间和落点。我们之前曾试图解析明白一个事情，简单游戏的无限排列组合玩法和变数让简单的游戏丝毫不会有懈怠感，反而有妙趣横生的娱乐魔力。这些循环往复和无限自由的组成都是挖掘玩家潜力的智慧元素，让经典的游戏不仅仅是需要付出时间成本的简单娱乐，更是一种对于象和形元素的领悟，并进阶到某种高度，使得用更复杂的描述可能无法界定，但能够使得用户们感知到复杂变换带来的冲击。这种冲击能带来玩家的愉悦。

当然这是一个靠相互感知支撑的世界，并不是每个开发者都能够同时洞见玩家潜在的真实需求的，也不可能每次都像Angry Birds那样能够直接借用德州五岁小男孩Ethan的创意关卡让自己看起来更亲和、更符合用户需求。相互感知借鉴可能被认为是可以接受的重构方式，甚至还能衍生出具有全球声誉的价值游戏，比如Halfbrick旗下的Fruit Ninja是在研究完苹果App Store前25强游戏之后构想出的力作，而Vince McDonnell旗下的僵尸农场（Zombie Farm）被认为是Farmville和Plants vs. Zombies完美糅合的结晶。



简单游戏和背后的理念努力

就像我们前文分析到的，简单只是这些成功游戏外在的形壳，类似Angry Birds和Cut The Rope能够获得GSMA、BAFTA和GDC这样专业奖项的首肯就意味着这些游戏总有被用户忽视的内在艺术价值。

它们最核心的共同点都在于简单闲趣，以最短的时间让用户从游戏中获得乐趣；关卡精练，能够满足用户零碎娱乐的需求；多关卡，有足够的价值吸引力引导用户持续回到游戏中；有相当的国际视野和思维力，让不同语言

和文化的区域用户都能够体验到同一游戏带来的快乐（共通的情趣和相似的信念，一样追求简单娱乐）。

当然任何游戏都需要玩家的意愿参与才能够最大化催化游戏的乐趣，好的体验在于整个进程而不在于繁复的功能设置，因此我们能够看到Angry Birds不会因为一个简单的发射动作而无聊，玩家对便捷通关的企图让每一次游戏都显得新鲜有趣。

用户情感是一条捆绑纽带

Chillingo公司的Joe Wee对游戏角色和玩家情感给出了一个清晰的纽带价值，他认为一个好的游戏形象或画面对玩家持续投入游戏至关重要（It's important to create characters users get emotionally attached to）。



Kim Daniel Arthur在谈到游戏价值的时候，也认为能够为用户带来值得回忆的体验就意味着是一款相当不俗的游戏。

Neil Young（Ngmoco）甚至还提到了要像做电影一样做游戏，把用户的各种体验感觉释放出来。

事实上我觉得用户体验已经超过了技术给予，Tiny Wings中那短翅膀的小鸟不懈的飞翔梦想让人深有感触。激发玩家打开心扉，引导玩家的情感融入在游戏粘性层面的价值远远比其他方式更有效，当然像辅助音效这种没有歌词的游戏音乐，能使得游戏更加传神和引人入胜。

而这些关键在于游戏环境的营造是否能够让玩家体会到他们对自己所属虚拟环境的拥有度和认同感，是否能够吸引玩家真切地关注游戏变化中的每一个环节，让玩家感受到这种变化与自己休戚相关。

声效同样具有小小的魔力

如果去掉Tiny Wings的声音或者在Angry Birds万圣节版里去掉声效的话，整体的游戏进程体验都将索然无味，声效所营造的是深入心扉的体验氛围，没有声效的张力，游戏进程也将缺乏表达力。

目前来讲很多小型的开发团队，对于声效环境并没有足够重视，认为通过工具和技术手段甚至外包可以处理好相关的问题。事实上我是觉得再小的团队如果没有属于自己的声效制作人总是不完整，外包服务也许能差强人意，但再好也没办法如自己一般理解到位。工具和技术手段确实可以带来不错的声效体验，但应该相信专业的音效师比工具和技术手段更理解音乐，而比起渗透开发者十足情感的音效，在捕捉用户灵魂和沉浸度方面具有的魔力而言，前两者都不过是一种敷衍行为。

类似Plants vs. Zombies这样的超级游戏，尽管刚开始只有四个人配合这个项目，但是在这四个人当中为声效环节保留了一个重要位置——主要负责Sound/Music的设计师Laura Shigihara。用户可以很明显地感受到Plants vs. Zombies中声效和整体的游戏进程和画面完美地融合在一起。

在声效的处理上，更为具体的环节可以涉及哪些类型的声效可能让玩家感到不快；对重复播放的声效，玩家的忍耐度（听觉疲劳）大体在什么位置；如何随机调整音调和节奏；如何利用音频压缩技巧来拓展有限空间的音效容量；如何配合整体的游戏进程设计恰如其分的音效氛围以及如何在合适的位置突出音效的魅力。



郑金条

游戏邦负责人，游戏邦主要关注和解析国内外社交游戏和手机游戏领域，并定期做深度行业阐述。

新技术、新平台、新境界

文 / 马宁

在这个夏天，微软最近推出了一系列新技术，产业升级的步伐也在逐渐加速。

第一个消息是Windows Phone 7.1 SDK，也就是Windows Phone Mango的SDK，虽然Mango的真机要到秋天才能上市，但让开发者提前了解Mango的新技术还是十分必要的。多语言输入法和多语言界面是Mango最有用的功能，除此之外，支持Socket将使得很多类似Kik的聊天工具将出现在Windows Phone上；Silverlight与XNA的整合也将使游戏开发更为简单；支持多任务，允许媒体播放、文件传输在后台运行；支持调用Camera，可以使条码和二维码识别类的软件大量出现。Live Tile也是一个有趣的功能，可以让软件的Tile动起来。如果说Windows Phone 7相当于iPhone 3GS的水平，那么Mango则赶上了iOS 4。而且，Windows Phone的应用数量、广告收入的增速也非常之快，当然这也得益于Windows Phone开发工具的易用性、Windows开发社区的强大。

第二个消息虽然只是一段视频，但Windows 8的名字会让全世界瞩目。Windows 8采用了与Windows Phone 7相同的Metro UI技术，从用户体验角度来说，Windows 8的Metro UI能够适应平板等触摸屏设备的要求。Metro UI最早被应用在Windows Media Center的界面中，2006年媒体播放器Zune采用了Metro作为Shell的UI风格。Metro UI在Windows Phone 7中发扬光大，并且为人们所熟知。Metro UI的灵感来自于英国的地铁系统标识，采用高对比度的色块和文字，使用户可以在快速移动中迅速找到并看清标志牌上的提示。微软为此设计了一套名为Segoe WP的超大规格清晰字体。与iPhone、Android基于图标的显示风格不同，“简单、清晰、现代”是Metro UI的鲜明风格。非常有趣的是，Google的I/O大会也有意无意地采用了Metro风格。Metro UI会成为Windows 8的主题风格，这一点似乎毋庸置疑，但如何兼容已有程序和鼠标输入，将成为 Metro UI最大的挑战。

第三个消息则是Kinect for Windows SDK终于发布了。Kinect是历史上销售速度最快的消费类电子产品，早就有无数人想将其用于其他领域了。虚拟试衣间之类的Kinect应用层出不穷，也有开源社区提供了针对Kinect的USB驱动程序，比如OpenKinect等。微软从谏如流，推出了官方版的Kinect for Windows SDK，终于让广大开发者可以名正言顺地使用Kinect SDK了。Kinect SDK中包括了动作捕捉、语音识别等功能，可以跟踪两个玩家的动作，同时捕捉20个玩家躯干上的信息点，支持C#和C++语言开发。微软在基础技术研究、平台开发技术领域优势明显，但一直不擅长为技术寻找好的商业模式。栽好梧桐树，引得凤凰来，这是微软在Kinect上做的最明智的决定。Kinect被称为鼠标之后人机交互界面的又一次革命，使开发者通过肢体动作来控制各种设备，将彻底改变未来的交互方式。试想一下，我们以后可以通过挥手来切换电视的频道。

看起来，这些只是彼此独立的技术，但如果将这些看作一个巨大拼图的几块碎瓷片呢？PC已经成为一个成熟产业，而各种各样不同的新设备层出不穷，除了智能手机、平板电脑外，还有很多的设备等待发掘。但所有的计算机系统都会有两个关键元素：输入和输出。



马宁
微软最有价值专家，Windows Mobile开发者。



Facebook工程管理揭秘之五

技术型领导

文 / 黄易山

黄易山（Yishan Wong）

1997年毕业于卡内基-梅隆大学。2001年加入PayPal，曾任高级工程总监。2005—2010年在Facebook领导研发，在公司研发环境的建设上发挥了重要作用。

Facebook前工程总监黄易山撰写了一系列文章，很好地总结了Facebook卓越研发文化中的宝贵经验。本文是这一系列文章的第五篇，也是最后一篇。

何谓技术型领导

所有从外部聘用的管理人员包括技术部门负责人，都必须能够编写代码，并且要达到炉火纯青的地步。如果是一家技术公司，CEO也应如此。

现在有个误区就是认为编程不是高管或者经理的必备能力，仿佛只是一种花哨的打字形式。但其他专业化行业都不这样认为：银行业高管必须能够阅读资产负债表；汽车业高管则需要了解催化转换器等。

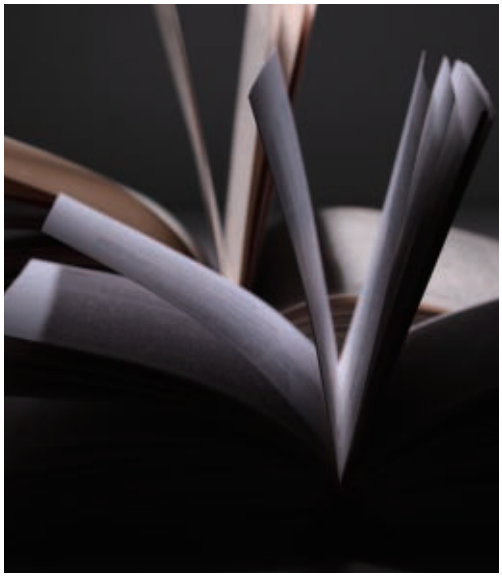
有人可能会说，技术的精通程度无法检验，因为一个杰出的管理候选人最近几年可能只关注于管理，与技术已无直接的接触。而且，一个杰出的经理可以管理一切事情。显然，这是不真实的。

当然，并不是希望候选人能用当前有限的扩展性技术创建一个大规模系统，或者在芯片集这种底层进行优化，或者能记住特定语言或框架的详细语法。但检验一个经理候选人是否具有较强的个人技术背景是合理并且可取的。当然我指的是基本技能测试，如果候选人曾经是一个称职的技术人员，他肯定能通过编程测试，包含某些简单迭代或递归算法，以及计算机基础学科中指针、散列和操作系统原理等概念题。

即使是一些门槛很低、许多人可能认为任何一个程序员都会的问题，还是有很多程序员搞不定（我并不是说能够做到这一点就意味着是一个优秀的程序员，但做不到这一点则意味着你肯定不是一个优秀的程序员）。在其职业生涯早期，他们发现自己不是优秀的程序员，但又恰好处在一个技术要求没那么严谨的组织中，因此他们能够被提拔，完全是因为他们碰巧很擅长与人打交道（或善于用人）。现在，他们中的许多人已经进入了技术管理和高管候选人的行列。此外，他们通常非常善于谈论一场精彩的比赛，听起来就像他们知道自己在做什么（否则他们也不会到那个位置）。

检验一个候选人是否具备技术实力的唯一方法是：给他们出一些简单的代码题目进行测试或者找一些他们写过的开源代码直接评估检验。不能通过测试或者没有可供验证的公开技术记录的候选人将不会被雇用。

原因是显而易见的——那就是管理者需要纵观大势，以便作出明智的决定。一个有经验但无技术背景的经理可能会有好想法，但在同等情况下，一个有类似技术背景的经理则可能有更突出的表现。换句话说，前者肯定提供不了技术领导力，如果希望你的公司成为行业的技术领导者，你的领导者首先需要具备技术。



为什么需要技术型领导？

一个没有技术型领导的“技术”公司往往会失败，原因可以归咎于以下两者或者其中之一。

■ 领导无法分辨技术人员执行的工作是否符合标准，因为在面临技术挑战时他们无法区分是技术人员执行力太差还是确实遇到了技术瓶颈。进而，也就无法实行绩效管理，这会导致业绩平庸，并将最终导致彻底甚至反复的失败。

■ 业务需求导致领导不顾技术人员的建议或者想法。当今严酷的商业环境要求企业领导推进企业不停地超越旧边界，这意味着领导不仅要告诉他的员工警惕“该死的鱼雷”，还要能够深化拓展，不能仅求安逸。不幸的是，非技术型领导人没有个人能力来衡量首要技术问题的实际风险状况（例如：某些特殊情况下已经非常过时的限制），并往往会推翻那些不应该被推翻的建议。

在Facebook之外，我见证了不止一个由于管理层缺乏核心技术力而导致的大型公司的失败。而在Facebook，个人技术能力恰巧是所有工程管理人员所必需的，甚至包括部门领导及CEO（是的，Mark Zuckerberg还在继续参与Hackathon编程活动）。这使得该公司敢于多次进行技术冒险，以达到更大的产品创新目标并实现一贯快速的前进步伐，正所谓越了解游戏规则，玩得就越好。

厂商新闻

IBM IMPACT 2011高峰论坛在京举行

6月8日，以“敏捷 转变 成长”为主题的IBM IMPACT 2011高峰论坛在北京举行。会上发表了《高速工作、高效工作、智慧工作》白皮书，全面升级业务流程管理系列产品并发布WebSphere的最新软件WebSphere Application Server 8.0。此次论坛覆盖了金融、电信、政府、电子商务等领域的端到端的行业解决方案和丰富的客户成功案例。

首届UC创新大会举行

6月8日，UC举办了以“优视共赢 创新无线”为主题的创新大会，与来自移动互联网行业的近千名嘉宾，展开了一场寻找移动互联网行业创新动力之源的创新之旅。会上，UC推出了中国首个自主研发的浏览器内核U3，并介绍了其移动互联网社交平台UC乐园一年以来取得的成果。

开放数据中心联盟发布应用模型路线图1.0版

6月15日，以推动开放及支持互操作的云解决方案为使命的开放数据中心联盟ODCA在北京宣布了其里程碑式的重要进展。首批公开发布的文件包括有8个定义了云部署IT需求的开放数据中心应用模型，以及一个针对云计算的开放数据中心愿景。这些成果为实现互通、灵活和高效的云计算愿景奠定了基础。

首届Esri中国开发者大会开幕

6月16~17日，2011 Esri中国开发者大会在北京举行。这是GIS技术提供商Esri面向国内GIS开发者以及IT专业人员举办的首届开发者大会。大会对ArcGIS应用开发技术进行了全面、系统的讲解与展示。同时，也展现了GIS技术与当前主流IT技术融合的绚彩魅力。

第四届SOA中国论坛发布软件平台研究白皮书

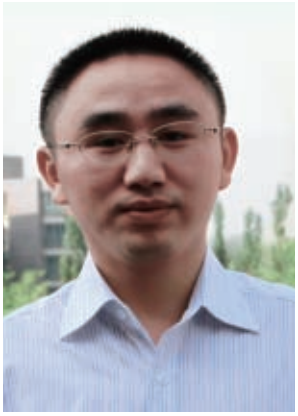
6月17日，主题为“SOA 平台到云”的第四届SOA中国论坛在北京举行。会上发布了由海比研究和普元软件联合研究、国资委研究中心企业所管理咨询部特别指导的《CIO：失控与可控——先进企业IT建设“软件平台模式”的经验与启示》白皮书，引起了与会者的普遍关注。报告指出，软件平台化使大型企业IT建设从失控走向可控。

“创新中心Jazz之旅”活动举办

6月22日，“创新中心Jazz之旅”活动在IBM中国创新中心举行。活动中，IBM公布了2011年Jazz创新大赛的最新进展和中期获奖情况，宣布了计划打造健康行业生态环境的IBM Rational“慧谷群英计划”，并推出了全新协同生命周期管理解决方案——CLM3.0。

红帽公司报告第一季度财务成绩

6月22日，红帽公布了其2012财年第一季度财报。报告显示，季度总收益2.647亿美元，订阅收入2.255亿美元，分别较去年同期增长27%和26%。红帽公司总裁兼首席执行官卫赫士将财务增长归结于云计算和虚拟化的发展。



谢志华
用友软件高级副总裁，1998年大学毕业后加盟用友，从普通程序员做起，现在是超过1500人的用友软件研发团队的掌门人。

解读光谱化开发模式

——专访用友软件高级副总裁谢志华

记者 / 付江

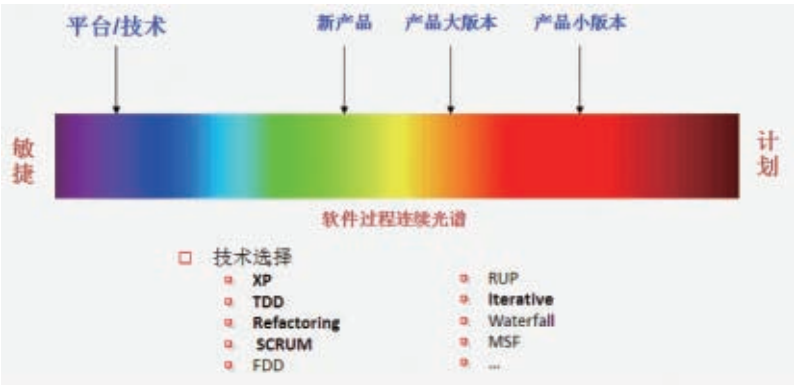
对于大多数软件产品而言，市场竞争都已经到了红海阶段。研发效率的提升从未像今天这般重要和紧急，而决定效率的是研发模式和流程。因此，记者对谢志华的采访也从这些内容开始谈起。

解读光谱开发模式

长期以来，用友一直在不断总结自己的开发模式和流程体系，走过了一条非常有特色的路线，“业界诸多经典开发模式，例如瀑布式开发、敏捷方法论、CMMI、IPD等，其精髓都被用友研发团队收归囊中，为己所用”。在谢志华看来，用友（研发）对先进理念一直坚持的原则便是：不照搬全抄、硬套，而是取其精神。“业界任何一个成大气候的软件公司都有自己独特的开发模式和流程管理，用友当然也不例外”，一谈到研发管理，谢志华的话匣子就完全打开了。

在他看来，业界很多经典开发模式，究其出处，都是为了适应某种特定场景下某一特征的产品开发而出现的，并不能完全照搬套用到具体的软件开发项目中，尤其像用友这种产品线复杂的研发体系，既有近乎于操作系统级别的底层平台级（UAP）开发，也有像财务系统、供应链管理系统等ERP产品开发，还有为不同行业和大客户定制的解决方案开发。对于不同类型的软件开发，模式选择是不同的。

图1　用友的光谱开发模式



谢志华认为所有的软件开发过程都处于一个软件过程连续光谱的某个位置，敏捷在一端，计划驱动在另一端，亦即“光谱”开发模式。某个具体的软件开发过程在光谱中的位置要根据它的主要特点和总体策略来确定。用友任何一款软件产品的开发立项都可以对应在光谱的某一个区间上。总体来说，平台级产品多偏向敏捷，小版本产品更偏向计划。

敏捷过程的特点是相对轻量级和松散式，主要依赖团队成员的知识和能力，注重现在的实现。而计划驱动则属于较重量级，规范程度高，强调过程和文档，而不是强调团队成员的能力。

产品选择哪种软件开发过程，可以通过两阶段的方式来确定。首先，确定在过程连续光谱中的位置，并选择一种过程风格，作为开发过程的总体指导框架。然后定制并配置这个过程框架，让它包含来自其他过程风格的技术，这样得到的开发过程将在敏捷和计划驱动技术之间取得平衡，反映出其在过程连续光谱中所处的位置。

谢志华强调，即使同一个版本的产品，在不同阶段也会处在光谱的不同位置。在做

概念与架构设计时，计划驱动的成分比较多，形式上主要表现为开发阶段划分明显，每个阶段需要详细而慎重的规划与推演以及评审。而到了产品具体实现阶段，就偏向敏捷了，Scrum、特性驱动（FDD）等被普遍应用。

汲取IPD+CMMI+ISO9000精髓

用友软件目前采用的是IPD+CMMI统一过程体系的开发模式。在谈到单一模式的利弊时，谢志华认为，如果单一采用ISO9000质量标准，这种“事后”再去检查文档是否齐全的方法将使开发过程完全不受控制，而CMMI的优势正在于强化对过程和成熟度的把控，他坦言：“CMMI对用友开发流程的改进是很有帮助的，提高了产品开发的质量管控和稳定性，使得每一个版本的成熟度和开发过程都比较稳定”，而CMMI的劣势在于缺乏面向市场的概念和快速反应的能力。

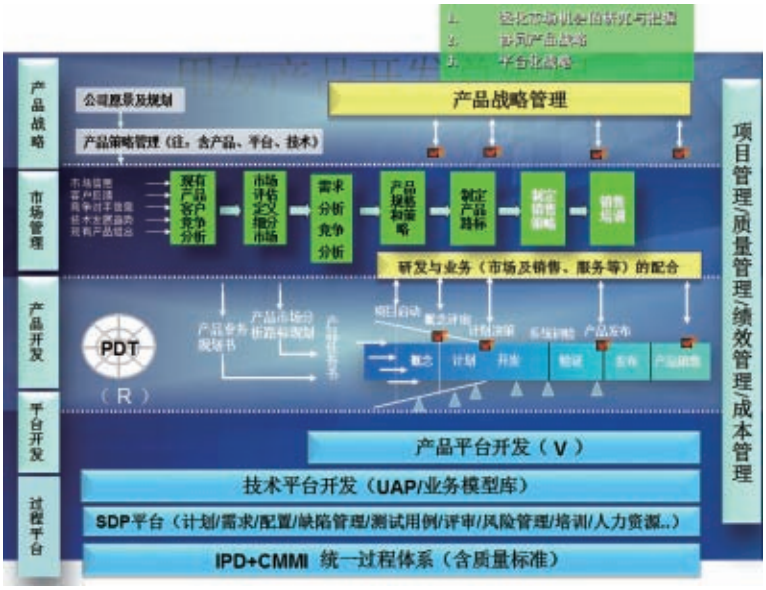
在图2所示的用友产品开发流程中，最上层由用友产品管理委员会负责公司愿景、规划及产品策略管理（含产品、平台、技术），新产品研发流程基于IPD+CMMI统一过程体系为指导，其中PDT作为重量级的跨部门团队，在产品概念阶段便正式组建，从概念阶段一直到发布阶段都执行IPD流程。产品立项强化对市场机会的研究和把握，注重跨部门、扩系统协同。此外，高端和中端产品都基于用友UAP公共平台，UAP在整合应用、快速开发、智能监控、个性化应用等方面起了至关重要的作用，用友大部分产品都架构在这个平台基础之上。

据悉，用友在2006年左右引入了IPD的思路，并在集团公司内普遍推广。基于这一理念，将新产品的开发看作是一项投资决策，基于市场、跨部门、跨系统的协同，同时注重产品的重用性和结构化流程。谢志华强调，当今大型软件产品的开发光有程序员是远远不够的，营销和市场人员，包括客户，都应该是产品开发过程的一部分。

鼓励开发人员创新

近些年来渐渐流行的敏捷方法论代表了

图2　用友产品开发总流程



一定的研发趋势，首先就是要充分认可和鼓励研发人员的创新精神。谢志华说，用友未来不会去做软件工厂，不是生产代码，而是研发产品，一个伟大的产品必然和一个富有创新精神的伟大团队密不可分，产品再好，团队不在了，灵魂也就没有了。在过去，传统的开发工程师只要按（功能）设计来执行就好了，但现在开发人员也已经被纳入到设计流程中，并鼓励他们进行创新。

对于新入职的技术人员，谢志华建议，如果是技术出身，应该从基础的专业工作做起，至少沉下心来做3~5年的专业工作，根据这几年的个人兴趣、特长和发展状况再决定下一步该如何走。这时会面临几种选择：进入研发管理岗带团队，或者继续走专业路线，甚至有一部分和客户相处能力比较强的，也可以转做客户经营相关工作。

采访到最后，当记者把“在用友研发团队这13年的最大心得是什么”抛给他时，他认真地说：“不断的自我学习。或许别人可以给你一些指导，但更重要的是自我学习和成长。”他经常建议员工，每天、每个礼拜，在回家的路上，有空的时候，想想还有哪些方面可以改进，并在接下来的行动中落实。坚持半年，就会发现跟半年前不一样了。

详解互联网产品开发中的“快”字诀

文 / 王晶

作者从产品、运营、技术、管理四个方面，诠释了腾讯互联网产品研发中贯彻的价值观——“快”。

当今互联网的发展，已不是大鱼吃小鱼的时代，而是快鱼吃慢鱼的时代。互联网产品的制胜原则就是一个字——“快”。在各种形态的产品研发中，我们始终贯彻如一的价值观之一就是“快”，我们应该如何来理解和诠释“快”？又会从哪些方面来执行贯彻这个原则呢？

快速迭代，快做快发

互联网产品不同于传统软件开发，我们面对的是上亿用户这样一个庞大的使用群体，他们是谁，有什么喜好，有何种习惯，会怎样使用我们的产品，是否喜欢我们的产品……

这些情况我们并不能准确地知道。因此，互联网产品的需求，并不能通过几个月的用户调研、市场调查、产品规划就能弄清楚，何况互联网的用户群体本身也处于飞速的动态发展之中。

那么，这种情况下如何发展我们的产品？如何对各种可能的产品特性做选择？用户将是最好的指南针，迅速让产品去感应用户需求，不断地升级进化，推陈出新，才是保持领先的唯一方式。要不断地倾听用户的反馈，不断地调整修改，然后决定你后面的方向。

所以，“快速迭代”是我们对产品的基本要求，能否做得足够快已成为衡量一款产品研发是否成熟的标准之一。以“QQ农牧场”为例，目前平均每天都会有至少一个版本发布，甚至更多，之所以能做到如此高的产品发布节奏，是由于我们一直坚持在做两件事情。

图1　QQ农牧场的“极速模型”



以稳定迭代，小步快跑

首先，QQ农牧场采用了一种有特色的敏捷迭代开发模式，我们称之为“极速模型”。

QQ农牧场的研发团队，由多个角色组成，包括：项目经理、产品、UE设计、前台开发、后台开发、测试、运维。以一周为一个固定的迭代开发周期，这一周时间包括了团队一次完整的各个角色的研发协作过程：迭代前有特性规划、迭代后有回顾，其中迭代过程也会包括迭代规划、开发、测试、发布等过程。但与Scrum敏捷迭代最大的不同是：并非在迭代结束时进行交付，而是能够在一次迭代中完成多次交付和发布过程。

此种方式看似简单，但其实对团队的综合研发能力是一个巨大的挑战。其中主要挑战来自以下几个方面。

- 特性需要能裂解成很细小的可交付的子特性，通常不超过两天的开发工作量。
- 迭代前，特性规划、沟通确认、界面交互及视觉设计这些工作均需提前安排完成。
- 迭代计划及评估过程，还必须考虑到特性/子特性之间的耦合关系以及开发人力的耦合关

系，合理地作出计划安排，保证开发过程的顺利进行，降低风险。

■ 要求团队成员工作咬合能力高，自运转能力高，需要长期默契配合。前台开发、后台开发、测试人员都能够高效率沟通，顺畅协作。

以特性为中心，随做随发

其次，我们产品研发的所有活动，都是以特性为中心开展的。一种比较通常的方式是规划一批特性，然后经过一个开发阶段进入测试，集中测试回归后完成发布。但在“QQ农牧场”，从特性规划、计划、开发、测试、发布都是以特性为单位来驱动的。也就是说当完成了一个特性的开发后，即刻转入测试、完成测试后即刻发布。在一个迭代周期内，会有很多不同的特性独立并行于从开发到发布的过程。

这还必须依赖于产品技术架构、测试自动化、运维发布自动化能力做支撑。但是“以特性为中心、随做随发”的核心思想，是产品、技术、项目管理、运维的指导原则，它是让产品的整个研发配套能力建设围绕这个中心来持续开展的基础。

反馈及时，响应快速

做到产品的快速发布只是第一步，其根本目的就是让用户尽快用到新功能，尽快得到用户反馈信息，以便及时地对产品开发做调整。所以，一个产品团队能否快速获取用户反馈、是否真正重视反馈并及时作出响应非常重要。经历了12年互联网的摸爬滚打，我们非常重视来自用户的反馈意见，并不断改进产品，积累了丰富的交付经验。

建设用户反馈渠道

首先，要解决如何搜集用户反馈的问题，满足不同用户习惯，提供多种方式的反馈渠道，让反馈及时得到。用户可以通过不同的渠道对使用的产品进行问题反馈，提出意见和建议。

重视反馈，快速响应

用户反馈、意见和建议就像一座矿山，为产品的发展提供了宝藏，但产品团队是否真正认识到它们的价值，是否能够快速地挖掘这些宝藏，却并不是一件容易的事情。

以QQMail为例，为了快速响应来自用户的反馈，在腾讯流传着一个1000/100/10的故事。

■ 每人每月必须回复1000条论坛用户帖子。

■ 每人每月必须查阅100篇与QQMail相关的网络评论文章。

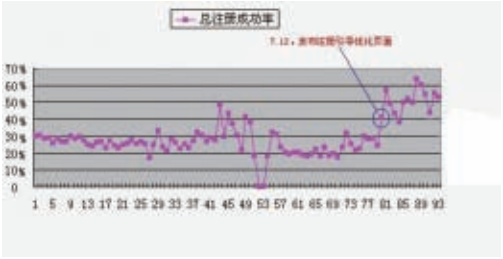
■ 每人每月必须处理10个用户反馈意见。

注重数据运营，有数据才有真相

无论事前经过多么细致的调研、多么缜密的规划，对于产品经理来说，一个新特性的发布，仍然是一个提心吊胆的经历：特性被用户的接受程度如何，用户将如何使用，新特性给产品带来了怎样的拉动或抑制，哪些特性可能存在交互、易用性、稳定性等问题。要想回答这些问题都很困难。

通过运营数据的分析，我们能够在短时间内获得对某个产品特性的准确评价，进而快速地指导产品下一步的发展。图2是一个产品93天内用户注册成功率的连续运营数据的例子。

图2　连续运营数据分析示例



从图2可以看出，7月12日前注册成功率稳定维持在20%~30%之间。7月12日对注册页面交互流程进行了优化并对外发布，之后两周的数据观察表明新的交互设计起到了预期的作用，注册成功率提升到了40%~60%，即使在7月17日、24日两天有定向向某省所有上线QQ用户发布消息时，其注册成功率也在40%左右浮动两个百分点。通过运营数据分析，能够快速地判断特性目标是否达到，进而指导下一步的行动。

快需要创新、需要实力

我们希望产品迭代得更快，但有了这个理

念就一定能够快起来吗？快不只是一种产品理念，更是一种技术实力，遵循着这个核心价值观，需要技术上的创新思维，让技术能力来支撑我们的“快”。

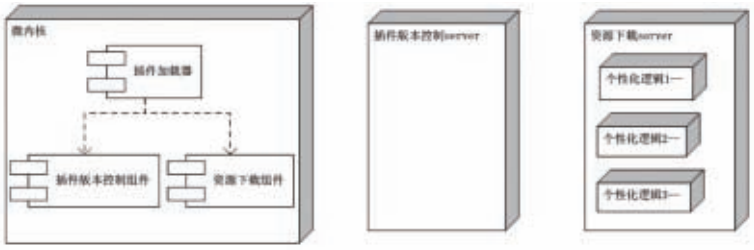
以QQ宠物为例，通过技术架构创新成功地提升了客户端产品的发布速度和更新频率。如果采用传统客户端方式的话，一次版本的全量升级需要6个月的时间，而在新架构下一次全量升级仅需1天。架构从以下几方面提升了快的能力。

客户端Web化技术：像B/S系统一样的开发方式和发布周期

有人会问：客户端的产品发布能快得起来吗？我们能做到让客户端像Web一样敏捷吗？答案是肯定的，我们的客户端微内核懒加载架构，将客户端Web化技术做到了像Web一样开发客户端产品。

整个架构由客户端的微内核、插件版本控制服务器和资源下载服务器构成，如图3所示。

图3　QQ宠物的技术架构



微内核简要介绍如下。

■ 整个客户端改造成为一个微内核插件平台，只有一个插件加载器、插件版本控制组件、资源下载组件。

■ 插件加载器，负责加载插件。

■ 插件版本控制组件，负责询问版本服务器获取加载的版本。

■ 资源下载组件，负责下载插件资源。

客户端的简要启动运行流程如下。

■ 获取版本：内核启动后，询问版本控制服务器，获取需要加载的版本。

■ 下载相应版本的XML配置。

■ 加载器解析XML配置。

■ 开始第一个插件加载逻辑。

■ 下载第一个插件的资源。

■ 加载第一个插件。

■ 继续加载子节点插件。

微内核懒加载架构与Web架构的比较如表1所示。

表1　微内核懒加载架构与Web架构的比较

	懒加载架构	Web架构
加载器	懒加载微内核	TT、QQBrowser、IE、Chrome、FireFox等浏览器
描述语言	XML	HTML
加载对象	插件	图片、视频、Flash等

同时，通过微内核懒加载架构还能做到特性即插即用，使产品灵活稳定。组件之间被强行解耦，大大降低了依赖性在联调、测试、系统集成方面带来的工作难度。由于每个组件都可以被独立下载，在客户端加载运行，这也就意味着发布风险的降低、效率的提升。

面向特性的竖向架构：以特性为开发粒度，提升开发效率

传统的产品技术架构多为横向的分层结构，而每一层又习惯于分配给不同的人来负责。这直接带来的一个问题是，我们以特性为粒度进行开发、联调、测试时会因为人员耦合、层耦合带来复杂性、引入风险。

图4　传统的横向分层产品技术架构



举个例子，比如开发一个login页面登录功能，可能需要Web前台工程师开发页面、Web后台工程师开发CGI、Server后台工程开发用户鉴权接口、数据库工程师做数据库表结构开发。那么这样一个简单的login功能，在联调、测试、发布方面就会牵扯很多的人力协作，而又因为每一层都需要改动代码，可能对这一层的其他功能代码造成影响。试问这样的方式能快得起来吗？

QQ宠物的新架构则以特性为中心，采用竖向的架构来解决这个问题，每个特性一个组

件，一个人负责开发，每个组件必须包括UI、逻辑、协议的代码实现。

图5　竖向产品技术架构



这样的方式，使得面向特性的开发模式得以强制化，从而提升了效率，加快了节奏。

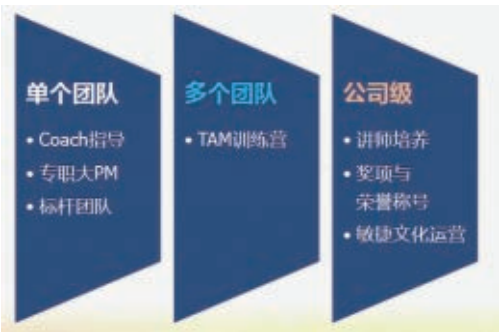
快需要手段

想快容易——做快难。在产品研发过程上，除了产品、运营、技术上的能力，我们还需要有必要的手段保证整个研发快起来。

Scrum敏捷开发：发扬光大

我们早在2005年就引入了敏捷开发，目前已经将Scrum结合我们自身的产品、文化、团队特点形成了自己的敏捷研发管理框架。经过自下而上的发展和腾讯人积极的探索和沉淀，逐步形成了经典迭代、极速、大象、运营这四个比较有特色的敏捷研发管理模式。而在敏捷的推广、实施方面，也已经有了一套以运营为理念的推广模式，把敏捷当作产品来运营，形成了“管理”、“工程”两条线，在多个维度推行敏捷。

图6　腾讯的Scrum敏捷开发



CI：持续集成，快速体验

CI在产品开发、测试阶段提升自动化效率方面非常有效。目前我们CI的发展水平还参差不齐，但从起初的自动编译已逐步加入了静态代码检测、单元测试、自动化部署等更多内容，开始为更多的研发团队所青睐。

作为加快产品发布的能力，CI在以下几个方面作用明显。

- 自动编译输出报告，维护代码可运行，及时暴露风险，降低集成成本。
- Dailybuild日构建系统，让产品经理、测试人员可以尽早进行体验和测试。
- 作为一个自动化系统，利用静态代码检查、单元测试报告等手段为团队提供报告，促进编码质量不断得到重视，降低缺陷解决成本、缩短解决时间。

灰度发布：提升发布的频率，降低发布风险

在互联网行业，灰度发布已经成为最重要的发布控制手段。有时我们希望通过向小部分用户开发新功能，让他们先来体验新功能、新特性。通过用户反馈、数据运营的手段及早获得反馈，及时改进。以此方式，既可以降低发布风险，也可以提升发布频率，加快发布节奏。

总结

快是一种追求、一种习惯，更是一种能力，这种能力需要产品、技术、运营、研发管理多方面的支撑才能够快得起来。这样的快，就像是中国的高铁，在高速的行驶中还必须让你感到安全、舒适、服务、便利。



王晶

腾讯R&D项目总监、敏捷教练。从事通信、互联网开发、项目及研发管理多年，目前负责腾讯多个业务线重要产品的项目管理工作，探索并推行适合腾讯的敏捷研发及项目管理。

责任编辑：董世晓（dongsx@csdn.net）

创业者如何打动投资者



李华兵
千淘资本创始合伙人

投资者所关注的创业者条件

在我眼中，创业者可以分为两种，一种是完全是从零开始的，一般是只有一个人、一个idea或者三五条枪，他们需要的是天使投资；另一种是可能已经具备一定规模，需要去扩大业务，他们需要的是风险投资。对于这两种投资要求的条件不太一样，但有些共性的地方。

首先也是最重要的是看团队的能力。只要团队是对的，即使最开始选择的方向不对，但经过一段时间的调整，该团队凭借其市场眼光、资本运作能力、产品把握能力，总能够找到合适机会，做成事业，投资者愿意给这样的团队投资。其中典型的代表是周鸿祎、陈一舟，他们虽然也走过一些弯路，但最后获得了成功。

在判断一个创业者是否可以投资时，还会看其履历，比如做过什么产品，带过多大团队，积累了多少用户，有哪些成功的样板等。虽然看起来比较经验主义，但这些东西会给投资者一种心理暗示，因为背景光鲜的人已经在某些方面获得了成功。比如许朝军，由于在盛大和人人网有丰富的资历，所以当他出来创业时，便轻松获得了投资者的青睐。

当然，如果没有这些光鲜的背景，但入手快，不只有idea，而且能拿出产品说话，执行能力强，同样也能获得投资者的认可，比如蔡文胜。其实投资也是很感性的，假如是在起点差不多的情况下，干得快的那个人一般会增强投资者的信心，从而获得投资。

其次，顺势而为，在正确的时间点做正确的事情。尽管上面提到最开始选择的方向未必准确，但也应该要尽量准确。比如现在要做一个Office软件或者操作系统的话，这明显是一个饱和的市场，不宜再涉足。再比如当年中国最早的B2C网站8848，相信任何一个互联网人都会对8848这个品牌予以崇高的尊重，因为它是一个里程碑，但事实很残忍，它成了先烈。失败的重要原因之一就在于对形势的判断失误：1999年时，中国网民的基数根本没有达到普及电子商务的层面，就像在一片根本就没有水、电这些配套的荒地里建摩天大楼一样，结果可想而知。

再次，投资者对创业者的创业激情更多的是一种感觉，比如创业者从一开始就每天去打高尔夫球、把公司搬到高档写字楼，这给人的感觉就是味道不对。在初期应该学会节约成本，并将精力全部倾注在事业上。此外，在创业早期，核心创业团队成员之间应该有合理的股份比例结构。投资者还会去考察创业者的领导能力、人品、口碑等，但凡有风吹草动，觉得对你这个人不太可信，比如偷过前公司的专利，那么就不会去投。因为有些东西是浸在骨子里的，只要基因是错的，将来可能就是错的。

最后，我想给创业者一些敬告。资本市场是一个残酷的战场，在同等条件下，把握住一次机会就会得到正向叠加，从成功走向更成功；当然，不要惧怕失败，倒了可以再爬起来。而如果坑了投资者的钱或者骗取了投资者的信任，那么就永远不会再拿到投资。所以要珍惜自己的信誉，否则以后这个市场上没有你的立足之地。



方天雨
麦包包副总经理

创业态度决定一切

到目前为止，单单拿着一个商业计划去找投资人并最终获得投资的故事，还是少数派。当然，一个完整的商业计划也是必要的，其中应该说明你的商业模型将怎样构建，未来怎样盈利。下面和大家分享一下麦包包在打动投资者方面的经验。

选对行业

找准方向并坚持下去，不断地优化和发现新的机会。用互联网语言来说就是“挺住意味着一切”。我之前在武汉做电子商务，叶海峰在嘉兴做箱包生产，通过交流发现，国内箱包市场没有品牌，而中国经济的发展又需要品牌箱包的出现，因此我们认为箱包品牌化将是一个潜在的巨大市场，便将它写入商业计划书，把箱包销售引入互联网。于是，通过这样一个切合点，我们开创了麦包包，开拓了一个非常有特色的电子商务模式。

建立团队

建立团队时最核心的问题就是寻找团队。团队的各成员应该认同团队的创业思路，具有共同的创业信念，就像当年中国足球队主教练米卢所说的“态度决定一切”。因为企业的发展不可能一帆风顺，在创业过程中会遇到各种各样的困难，所以我们比较看重这一点。也正因如此，到目前为止，我们高管团队的主动离职率为零，这让我们非常骄傲。

此外，创建团队时考虑的另一个重要问题就是各成员应该乐于分享。各成员应该分享他们的未来愿景，其实归根结底也是一个创业信念的问题。

经营团队

建立好团队后，就要管理好团队，我更偏向于说成是经营团队。经营团队的杀手锏就是企业文化。对投资者来说，企业文化是他们非常看重的一个方面，我们的企业文化也是成功打动投资方的利器。

企业文化应该在创业早期就制订好，而且越早越好，因为人的思维，不是一两天能改变的，如果企业做大后再制订就没那么容易了。因此从创业第一天开始，我们就非常注重企业文化的建设。同在浙江的阿里巴巴对企业文化的打造相当成功，可以称得上是我们的楷模。

选择是双向的

选择一家相互尊重和相互认可的投资者，这非常重要，可以说是我们选择投资者的标准。选对了投资者，它会在你公司还小的时候，不仅提供资金上面的帮助，而且还会在团队建设和管理、推荐合作伙伴、业务流程等方面给你很大的帮助。

作为在路上的先行者，我们已经打动投资者好几次了，每次的感受都不同。在早期初次融资时，发挥作用的是创业激情、一个好的方向和一个已经启动的团队；在后期再融资时，更多的是团队对投资者承诺的执行力。只有具备坚定的执行力，才能使投资者信心满满持续进行投资，有了这些保障，上市也便水到渠成了。



傅宇权
万兴软件高级副总裁

融资如“联姻”

市场上创业团队越来越多，市场上的投资资金也很多，如何让两者能够很好地匹配在一起？创业者如何拿到第一笔有价值的投资？当创业者有了融资的想法时，就必须先回答一些问题。

站在创业团队的角度需要回答：要钱做什么？计划多久花完？还需要什么资源？打算付出多大的成本（如股权）？愿意冒多大风险（如果对方提出对赌条款）？而站在投资者角度需要回答：产品（服务）是什么？客户是谁（能赚多大钱）？盈利模式是什么（怎么赚钱）？门槛有什么（能赚多久的钱）？资本退出的机会有多大（再融资或上市）？

作为创业项目，尤其是创新型项目，上述十个问题往往不会有很成熟、明了的答案。但创业者至少要给出明确的答案，让团队以及投资者有足够的信心。能够很好地回答上述问题，基本就算是比较好的项目了。

投资机构在不断寻找项目的同时，事实上是在不断寻找合适的“人”。投资投的是团队，而非仅仅项目本身。所以，想清楚上述十个问题之后，我们必须回顾自己的团队，是否具有以下特征：足够高的行业视野，以及敏锐的视角； 足够强的创业动力，以及坚韧的精神； 足够好的运营经验，以及团队的意识。

有了好的项目，再有了以上团队特征，众多投资机构基本会趋之若鹜。接下来，就是我们应该如何选择投资机构了。

融资如“联姻”。婚姻，不是为了对方的财富或美貌，而是为了生命基因的延续。融资也是如此，不仅仅是为了资金，更多的是为了通过资金及投资机构的行业背景获得更多的资源，以利于项目的长期良性发展。因此，在选择投资机构时，要根据自己的资源需求选择合适的机构，也要对投资机构未来可能派驻的代理人有足够的了解，以利于未来的长期合作共事。

选择了有意向的投资机构后，则进入实质性商洽阶段。在谈判过程中，一般重点会围绕以下几个方面进行商洽：投资机构进入的基准条件，可能包括投资机构的资金来源、项目团队的历史沿革、业绩目标、治理架构等；投资机构未来的退出方式及对应的条件，可能包括对赌条款（如果是成熟项目，不建议设立对赌条款）、再融资（包括IPO）条件等，视具体项目而定；项目估值，项目估值与双方愿意承担的风险及期望的收益直接有关，一般可参考市场上同行业通常的估值方式和估值标准，不过，互联网项目的估值会比较难把握；未来的治理架构及决策模式。

依据上面提到的经验，我们在过去两年内进行了两轮融资，第一轮以引入资金并规范企业治理架构为目的，第二轮以引入资金及产业资源为目的，基本达到了预期。



王毅
适合我旅游团购网创始人

在校大学生创业的成功心得

适合我旅游团购网由我们清华大学三位在校学生创办，目前已获得天使投资，成为浩荡团购大军中的一颗新星。下面为大家分享一下我们是如何获得第一笔投资的。

我始终认为，要获得投资的最关键一点，就是需要一个好的团队。“投资投的不是点子，而是团队”，这句话是非常有道理的，没有好的团队，不管多么美好的想法最终都只会是幻想。那么，怎样的团队才算是一个出色的团队呢？

每位成员都有一颗创业的心，非常喜欢他们目前所做的事情，即使没有经费，也愿意自己掏钱继续创业。如果团队中的成员感觉自己做的事情比其他成员多，就觉得不舒服，像吃亏了似的，这样的团队迟早会死掉。一个好的团队，每位成员应该以做得多

而感到自豪，因为团队是在为自己创造，让自己的创造被千千万万的用户使用，对程序员来说，这是多么美妙的一件事情啊！与此同时，团队成员之间一定要相互信任，相信对方也是在全身心的创业，也会把自己的工作做好，不能因为每个人做的不同，就感觉其他成员在偷懒。

专注，既然选择了创业，对待其他事情，就应该抱着“能放则放，能缓则缓”的态度，这不仅是对投资者负责任，更是对自己负责任。同样，业精于专，团队的想法，最好也专注于一点，例如现在团购非常火爆，也日趋成熟，但是我们的团队只专注在旅游上，其他方面我们不涉足，别人挣钱我们高兴，但我们只把有限的精力放在在线旅游电子商务上。

执行力强，有了好的点子，需要立即实现，不能拖拖拉拉，发现了问题，需要立即更正，不能互相推卸。当然，在决定上，最好是创业团队所有的成员一起进行，这样才更具有凝聚力。

除了好的团队，剩下的就是一个好的点子了。好的点子不一定是最新的，例如在Facebook出现之前，已经有过类似的社交网络；但是做出来的产品一定要注重用户体验，并且持之以恒地坚持下来，“故不积跬步，无以至千里”，只要积累够了，自然能“厚积薄发”，投资者自然会找上门来。

当然，我们也有一些做得不足的地方，例如，我们三位创始人都是在校学生，需要处理好创业与课程学习在时间方面的冲突，虽然暂时把课程学习的重要性放在创业后面，但毕竟也占用了不少的时间，这一点曾成为投资者最大的担心。

要想把在线旅游团购做大做强，还有很长的路要走，我们将会用自己的行动，使适合我旅游网得到健康、蓬勃的发展。



卢伟
深圳双子鱼科技有限公司创始人

有的放矢地推销自己

由于看好移动互联网产业的发展，我辞掉在日本的高薪工作，结束了7年的海外生活，回到中国和几个志同道合的朋友组建了现在这家智能手机平台的软件开发公司。比起其他单靠自己开发、根据在App Store上的下载量来获利的开发者和开发公司来说，我觉得自己幸运得多。因为日本的移动互联网市场发展早于中国市场，在回国之前我已经和日本的多家客户谈好了外包项目，所以回国后我们一边承接日本的外包业务，一边尝试着开发自主知识产权的产品。由于有国外客户的支持，到目前公司也未出现捉襟见肘的局面。但是像我们这样的创业型公司如何给自己定位、如何发展壮大，一直是我思考的问题。

公司要发展，单靠公司几个人的单打独斗是不够的。能够获得风险投资家的青睐，那是求之不得的事情。在公司成立的八个月时间内，我也接触过一些投资者。有些投资者比较注重公司的产品是否是行业第一，目前有多少活跃用户等；有的注重公司的财务；有的注重公司团队的实力。所以我觉得在考虑如何吸引投资者前必须做好两件事：一是要了解你的优势在哪里，技术、推广还是产品；二是希望得到投资方哪方面的支持，财务的还是市场推广资源的。只有在了解了这些基本情况，才能有的放矢，更好地推销自己。

另外，就我们自己的情况，目前以承接外包项目为主，卖点就是服务，所以把产品质量和服务质量定为公司的生命线，重视团队的技术培养。最近，一家提供过服务的日本公司决定注资我们公司，这是我预料之外的。所以我觉得，把眼前该做好的事情做好了，机会说不定就在身边。

提高团队协作能力是管理的核心所在

——亚太软件研发团队管理峰会上海站即将召开

记者 / 陈秋歌

伴随国内软件行业的迅猛发展，企业间的竞争日趋白热化。企业面临的最大挑战不是资金的缺乏，而是企业高端软件研发管理人才的缺失。高素质的软件研发管理人才是企业发展的领军人物，是企业获胜的核心竞争力。迅速培养高素质的软件管理人才，是每个企业迫在眉睫的首要任务。

管理人才紧缺且难培养

一个好的技术领导者不但可以带领“部下”百战百胜，甚至可以让一个企业“起死回生”。然而正所谓“千军易得，一将难求”。当前国内中高端技术管理人才不仅仅是紧缺，而是严重供求失衡。

对此，亚太软件研发团队管理峰会组委会副主席魏立春分析称：“由于IT业发展水平逐步提高，对IT业的管理和技术开发上也提出了更高的要求，此时的高端人才不但要求具备IT方面的技能，同时还要具有宏观把握能力、创新性思维及行业相关经验，又很难成批培养，这是造成高端人才短缺的主要原因之一。”

而企业自己培养管理人才，也面临着种种困难。正如魏立春在接受采访时说的那样，技术管理人才是实践和能力的综合体，培养周期长、可复制性差，且跳槽频率高，公司间挖人现象严重。一旦被挖走，企业之前的努力就会功亏一篑。

于是很多企业便把眼光投向了类似于麦思博这样的以“分享卓越软件研发管理实践”为核心理念的技术管理培训机构。很多企业凭借实际有效的培训转化，获得了很好的效果，企业也逐渐认识到优秀的技术管理培训是帮助企业实现成功的一个很关键的途径，麦思博CEO刘付强表示。

提高团队协作能力乃管理的核心理念

企业要获得较好发展，除了要有一个好的管理者外，还要有一个高效的开发团队。刘付强深有感触地说：“其实解决企业的管理瓶颈并非某个人的问题，更多的是整个团队的问题。一个技术企业在发展过程中遇到的问题往往是环环相扣的。”所以提高团队整理协作能力才是管理的核心所在。

技术管理者如何建设高效研发团队？刘付强认为，软件研发团队的成长更多的是依靠“智”，即智慧、智能。通过培训驱动、升级并改变和提高整个团队的智慧、思想和管理水平以帮助企业更好的发展，是麦思博总结出的“以智驱动”的培训模式。与世界知名软件企业的精英们一起探讨问题、分享经验，获得更多研发智慧，是每个企业所津津乐道的事，金山软件研发团队推荐称：“亚太软件研发团队管理峰会（msup open day：即MPD大会）恰恰提供了这样的交流平台。”

MPD大会是麦思博为企业打造的学习、交流、分享平台，主要面向于全国软件研发主管、开发经理、架构师、技术总监等有影响力的技术管理者；大会按软件研发中心的岗位职能划分，分为产品经理、团队经理、架构师、开发经理、测试经理五大会场；并携手世界级一线软件研发团队带头人，通过workshop分享他们的卓越团队管理和开发实践攻略。

“大会主题明确，内容务实。期间，没有商学院式的夸夸其谈，有的是很多实际的案例分享。作为听众，我们同时了解了国际公司和本土公司的研发经验，相互印证，受益良多。”爱立信XLTT部张经理如此评价MPD大会。

新一届MPD大会即将登陆上海，据悉本次大会根据上届客户的反馈进行了很大的调整，更加贴近客户需求。具体有多精彩，让我们拭目以待吧。

谨慎使用NoSQL

——知名数据库专家Raghu Ramakrishnan访谈

Raghu Ramakrishnan是Yahoo! Labs搜索与云计算首席科学家、副总裁，世界知名数据库专家，ACM和IEEE Fellow，曾任ACM SIGMOD主席，现任ACM SIGKDD执委。近日，《程序员》杂志对Raghu进行了访谈，对如何成为一名计算机科学家、技术者创业需要注意的问题以及云计算和数据库未来的发展方向等话题进行了广泛探讨。

成功的计算机科学家

《程序员》：请介绍一下您的学术和工作经历。

Raghu：我从1987年开始在威斯康星大学麦迪逊分校的计算机科学部数据库系统研究组工作。1999年时，我创建了一家公司（QUIQ），主要做一些类似于现在Yahoo!问答、Quora这样的问答系统。尽管我是一名大学教授，但我还是很喜欢工业界这边的工作。我的研究方向主要是网络数据管理，而这也是Google、Yahoo!以及Microsoft这些业界巨头的研究热点，因此在2006年，我和这几家公司进行了接触。

之后我进入Yahoo!，开始了我在Yahoo!的研究工作。我们的研究团队在超大规模数据处理、云计算这些方面做了广泛而深入的研究。我们的研究成果不但在Yahoo!得到了广泛应用，而且也对数据库系统研究做出了不小的贡献。因此，我非常喜欢Yahoo!的这份工作。

《程序员》：作为一位成功的计算机科学家，请给有志成为计算机科学家的年轻学生一些建议。

Raghu：作为一个研究工作者，如果你所工作的领域已经相当成熟的话，那么你很难做出研究成果。但如果你所工作的领域还处在起步阶段的话，那么情况就会大不一样。一方面，在新领域研究出成果要相对容易不少；另一方面，当你在一个崭新的领域里做出重要的发现后，其他学者将会以你的研究为基础，进行进一步的研究，从而使你的研究成果具有更大的影响力。

《程序员》：对计算机专业的学生来说，您认为哪些理论知识是必需的呢？

Raghu：我们现在生活在一个数据大爆炸的时代，无论是互联网还是现实生活中，每时每刻都有大量的数据产生。如果我们有效地利用这些数据，我们的工作和生活将会发生革命性的变化。因此从某种意义上来说，数据就是一切。

为了高效存储海量数据，我们需要更好的数据库技术；为了从这些数据中挖掘出有意义的内容，机器学习这方面的知识不可或缺。拿生物学打个比方，我认为生物学将会和计算机科学走得越来越近，因为生物学的研究涉及了越来越多的计算机知识，生物学家需要用计算机推导、建立和分析他们所需要的模型。因此我觉得未来的生物学家将会同时持有生物学和计算机科学两个学位。鉴于计算机科学家面临着越来越多的数据分析和数据处理的工作，我认为未来的计算机科学家，需要具备计算机科学和统计学这两个学位。

技术出身的创业者

《程序员》：您是QUIQ的创建者，您从这段创业经历中得到的最宝贵的经验是什么呢？

Raghu：如果钱到了手里，千万别急着花出去（笑）。

《程序员》：请您给新技术创业者一些建议吧。

Raghu：对于创业者，尤其是那些身为技术人员的创业者来说，他们有一个很自然的倾向，就是过分重视技术。当然我并不是说技术不重

要。我想表达的是，作为一个创业者，集中精力找到并处理正确的业务才是关键，因为这关系到公司的利润和成败。技术只是解决问题的手段，而问题本身才是重点。找到正确的问题，并使用更好的技术来解决它，才是成功的关键。

《程序员》：QUIQ是一个在线问答系统。现在涌现了诸如Quora、Stackoverflow这样的新型在线问答服务，您如何看待它们？

Raghu：我只能说QUIQ做得太早了（笑）。

相对于单纯的在线问答系统，我更关注群包（Crowd-sourcing）系统的发展。实际上，在线问答只是群包系统的一种特定形式而已。我认为群包系统在未来会变得越来越重要。

群包系统的主旨是发动网民的力量来解决问题，这使我们可以轻易解决一些以前难以处理的问题。现在已经有很多这样的系统，Yahoo!问答和Quora就是两个群包系统的典型。我认为未来这样的系统会越来越多。

云计算和数据库系统的大师

《程序员》：您认为云计算和先前技术之间的本质区别是什么呢？

Raghu：在回答之前，我先讲点过去的事情。2000年时一种称之为ASP（应用服务提供商）的商业模式刚刚兴起。简单来说，ASP允许你“购买”基础设施，你可以把你的应用安装到别人的服务器上运行。一些公司还提供了另一种形式的服务：提供一个模板应用，你可以在这个模板应用的基础上定制你自己的应用。Salesforce.com就是早期采用这种模式的公司之一。

以我创建的QUIQ为例，我们主要做在线问答系统，客户包括COMPAQ、SUN和国家仪器这样的大公司。这些公司有大量的客户支持服务是通过问答系统来完成的，通过QUIQ的服务，它们可以把问答系统部署在QUIQ的服务器上运行。换个角度说，IBM曾经推出过一种网格计算技术，叫做Load Leveler，它允许你同时使用大量的计算机，从而在短时间内完成计算任务。

所以我认为云计算实际上是早期理念的一种融合。2000年那会儿，没几个公司会真的使用ASP模式。我现在还记得，当时我们去COMPAQ推销我们的产品，我们说“我们不但可以完成你们的客户支持系统，而且还可以在我们的服务器上运行这些服务”，而他们认为我们在开玩笑。我们花了很长的时间才说服他们。

随着云计算概念大行其道，人们逐渐接受了诸如租用服务这样的理念。所以我认为，从商业角度上来讲，云计算应用模式和过去的应用模式并没有本质的区别，如果非要说区别的话，也只



CSDN暨《程序员》总编刘江专访知名数据库专家Raghu Ramakrishnan

是概念的普及程度和接受程度不同而已。而从技术角度来讲，云计算和之前的技术还是有一些差异的。从学术的角度上来看，云计算涉及到很多有意思的研究工作。

作为ACM SIGMOD（注：Special Interest Group on Management of Data，ACM在数据库方向上开设的顶级会议）的主席，ACM邀请我和ACM SOSP（注：Symposium on Operating Systems Principles，ACM在操作系统方向上开设的顶级会议）的主席共同成立一个云计算的会议，这就是SOCC（注：Symposium On Cloud Computing，ACM成立的云计算研讨会），足见学术界对云计算的重视程度。

总之，我认为云计算是一场正在发生的革命，在接下来的10年里，它会发展得越来越好。

《程序员》：您认为云计算会给计算机科学带来什么影响吗？

Raghu：云计算系统要求多用户入驻，并且灵活易用，这可不是那么容易做到的。为了实现这些特性，我们需要深入研究数据库系统、操作系统、虚拟化等技术。

由于云计算的广泛应用，对高性能网络连接的需求也会越来越大。而云计算环境的特殊性，使得诸如并发控制这类老问题有了新版本，而我们现在还没有什么好办法来应对它。总之，要实现高可用性、多用户入驻、高灵活性等特性，需要在很多方面进行深入研究。而数据库系统和操作系统，将会是受云计算影响最大的两个方向。

《程序员》：请谈谈您对NoSQL数据库的看法，以及NoSQL数据库和关系数据库的关系。

Raghu：一些NoSQL的鼓吹者认为，传统关系数据库提供的很多特性（比如事务和花哨的SQL语句）并不是必须的。老实说，我个人认为，NoSQL数据库的出现源于实际的应用需求。因为在当前的很多大规模数据应用中，容错性、高可用性、可伸缩性和可扩展性是不可或缺的，而传统的关系数据库很难支持这些特性。因此我们做了一个折中：牺牲ACID、事务来换取高可用性和更好的伸缩性，以更好的满足实际需求。

但你需要注意，折中意味着拿一个好东西去换另外一个好东西。有收益就会有损失。NoSQL数据库在一些应用场景下工作得不错，但从功能



Raghu Ramakrishnan在TUP Masters系列第四期“深入云计算实战”上做演讲

上来说，NoSQL数据库很难做到关系数据库系统那么全面，就像鱼与熊掌不可兼得。假设你在构建一个系统，你当然希望这个系统具备的优良特性越多越好。但问题在于，受当前技术的局限性，同时支持所有的特性是不可能的。NoSQL数据库提供了高可用性、灵活性、多用户入驻这些特性，关系数据库提供了结构化存储、复杂的查询处理能力，而你需要在其中做出取舍。

我认为，在未来的5年中，关系数据库将会变的越来越灵活，而NoSQL数据库的功能则会越来越强大，这是它们发展的必然趋势。

《程序员》：您对NoSQL数据库和SQL关系数据库的集成有什么建议吗？

Raghu：如果你打算使用NoSQL数据库，那一定要慎之又慎。

这听起来可能有点奇怪，毕竟我本人就是一个NoSQL数据库的研究者。之所以这么说，是因为我发现NoSQL数据库被很多开发者误用了。诚然，NoSQL数据库提供了一些关系数据库没有的优势，但如果你的应用不需要处理海量数据、不需要那么高的可用性和灵活性，那你为什么不去使用成熟的SQL数据库呢？难道你使用NoSQL数据库的唯一原因就是它很流行吗？

就算你确实需要灵活性和高可用性，你仍然需要仔细考虑，毕竟这些特性将会牺牲掉一部分的查询处理能力。因此我认为，与其去考虑如何集成NoSQL数据库和SQL数据库，不如花些时间好好想清楚你的真正需求，这才是重点所在。

（感谢巩朋对本文的支持。）

云计算实践分析

演讲人 / 李德毅

在第三届中国云计算大会上，李德毅院士就我国当前云计算的实践情况做了三方面的分析：云计算给软件园带来的变化、传统电信行业的纠结、云计算实践的试点。

全球云计算自2007年开始成为整个信息领域，尤其是通信、计算机和网络领域的前沿和热点。经过这几年的发展，云计算变得越来越聚焦、越来越务实。可以说，我们已经走过了“理念清晰、操作模糊、八仙过海、云里雾里”的一段路程，正在进入云计算快速发展的阶段，走上了云计算的应用之路，期盼着“云开雾散、腾云驾雾”的这一天早日到来。在这里，我想围绕我国当前的云计算实践作三方面的分析。

云计算给软件园带来的变化

截至2009年底，我国35个国家级软件园入驻企业达2万多家，从业人员超过130万人，占全国软件从业人员的一半；销售收入7677亿元，占我国软件产业总额的60%；软件出口额为75.4亿美元，占软件出口的40%；总收入与GDP的比值达到2.29%。可以说，软件园是我国软件产业中最重要的组成部分之一。那么，云计算会给软件园带来什么变化?

下面以一个软件园中的软件测试中心为例。目前，我国软件测试中心的主要工作包括以下几个方面：

- 作为第三方，面向各行各业的IT系统实施现场测试，为系统的验收和鉴定提供依据；
- 通过国家和行业的授权认证，提供软件质量保证的信誉担保；
- 对基于互联网的应用软件进行测试。

云计算下的软件正在转向服务，对于软件测试而言，前面两部分的工作会越来越少，第三部分的工作会越来越重要。互联网环境中的应用软件测试主要是测功能、测性能（如响应时间、负载能力等）、测安全性、测可靠性、测易用性、测互操作性（如数据、接口的兼容性等）以及检查文档的一致性和完整性等。为了完成这样的测试任务，传统的软件测试中心首先需要软件生产商提供测试任务列表、详细的文档和待测软件；然后，根据测试任务列表，软件测试中心制定相应的测试大纲和测试任务包，在搭建的物理测试环境中实施测试，最后返回测试结果。通过多次的回归测试，软件生产商逐步更正检测出的软件缺陷，改善软件产品的质量。

大家知道，建立物理的测试环境是个耗资、耗时、耗力的工作。应用软件的规模不



李德毅，中国电子学会副理事长，中国工程院院士，欧亚科学院院士，中国人工智能学会理事长，中国电子学会云计算专家委员会主任委员，国家973项目首席科学家，国家和军队信息化专家咨询委员会委员。

同，测试环境又会有很大差异，尤其是负载测试和压力测试。以前，一个针对互联网应用软件的压力测试的环境需要自行构造，如果是百万级的并发用户，数据流量大（可达每秒GB级），实现起来将会很难。有了云计算的方式，软件测试中心则可能转为通过云计算方法来租用软硬件设施及相应的服务，用来快速、低成本地模拟网络环境和用户流量。

举一个对社交网站MySpace进行百万数量级的并发用户压力测试的案例，这个案例来源于SOASTA CloudTest。传统的方法采用1:20的缩微仿真模拟5万并发用户，使用专业测试软件在每台PC上模拟1000个并发用户，需要50台PC和2台服务器作为测试控制器和结果数据库；而操作系统、数据库、测试软件等需自行安装和配置，网络环境需自行管理和维护。若使用云计算下的计算资源租用方式，如已经广泛应用的Amazon的EC2 Spot Large Instance，每个Large Instance（7.5GB的内存、4核虚拟CPU、850GB的存储空间和Linux操作系统）可模拟1300个并发用户，则800个Large Instance和2个Extra-large Instance作为测试控制器和结果数据库，完全可以模拟百万级的用户规模。与传统方法形成鲜明对比的是，这种方法的部署时间甚至可能控制在小时数量级上，而且租用价格也相对低廉，每个Large Instance每小时仅仅0.115美元，租用的所有Large Instance的费用每小时才100美元左右。

通过这个案例可以看出，基于云计算的方式不再需要自行物理配置和维护所需的软硬件和网络设施，准备时间短，成本低，按需租用，灵活方便，云计算给软件测试带来了极大的改变。同样，软件园区中其他应用领域的软件企业也会因为云计算而发生改变，如电子邮件服务、云存储、客户关系管理服务等。尤其突出的是，越来越多的移动计算需求，催生了基于云存储服务的视频点播、相片管理应用、移动办公等新服务。如果通过云计算为每个移动终端用户提供100GB的虚拟存储资源容量，相当于100部左右的高清电影或5万张左右的电子照片，在带宽迅速发展的前提下，避免了用户各自购买服务器的开销和各自管理存储设备的负担，将会极大地提高用户的移动服务体验。

软件园汇聚了众多软件企业和软硬件设备，因此，软件园往往也是一个城市的耗电大户。粗略地估算，一个入驻数百家企业、几千名员工的软件园，每月需支出的电费将会超过240万元。如果软件园的企业可以共享同一个云计算平台（包括硬件平台、开发工具、测试环境和运行平台等）来进行软件即服务、平台即服务和基础设施即服务的工作，将会减少软硬件的耗电需求，逐步构建起绿色的软件园，促进节能减排。

这些云计算所带来的软硬件的部署、开发等方面的深刻变化，会使软件园内小而散的行行色色的软件企业，通过动荡、分化、改组和归并，演化为有特色的云计算中心，针对细分需求，提供集中的、面向不同领域的计算、存储和交互服务，逐步形成社会化、集约化和专业化的面向特定领域的服务。当前，有些地方另起炉灶，成立云基地的做法，也许不是一个正确的选择。

传统电信行业的纠结

我们知道，北京固话用户每月21.6元的月租费持续了十年左右，一户一月21.6元，在北京、在全国的范围内，这对传统电信行业是一笔可观的收入。很多的消费者对此感到不满意，同时，固话用户的数量在全国范围内急剧减少。据工信部网站的统计数据，2010年1月至7月，全国固定电话用户累计减少了997.4万户。现在，更多的用户倾向于使用手机、电脑和互联网，曾经的固话业务通过WiFi、IP电话和互联网应用等途径流失了，传统的收费被绕过去了，对于传统电信行业而言，能不感到纠结吗？

从这种纠结可以看到，传统电信行业，曾经以提供话音为主，仅仅满足了大众用户最基本的需求。随着数据通信的普及、互联网以及Web应用的快速发展，人们对相互之间沟通的质量要求提高了，这导致了文字、话音、视频等多种多样的、更为直接和个性化的需求，因此，传统电信运营商的纠结，归根结底是新技术所带来的市场需求的提升，是需求的精细化造成的。

过去，电信行业常常是地区割据、行业

割据，是一种典型的金字塔式的层次结构。现在，要求我们从割据到融合。原来是先有设备制造商，再有电信运营商，以后再有增值服务商。我们曾经把非话业务叫做增值服务，现在，谁是谁的增值服务？说不清楚了。对于做广播电视的人来说，不但电视传输要IP化，而且在实现IPTV的同时也可以提供话音服务。因此，做广播电视的IPTV，认为话音是他们的增值业务不行吗？VoIP将声音的模拟信号以IP数据包在互联网上实时传递，利用全球互连的网络，可提供比传统话音业务更多、更灵活的服务。话音本身已成为一种互联网应用，从工信部网站上的统计数据可以看到，传统意义上的“非话”业务，其收入从2008年占电信业务总量的33.1%，到2010年就快速上升到了42.3%，

云计算是网络环境下计算资源的优化，是技术的创新，也是商业模式的进步。云计算的实践没有现成的成熟模式可供借鉴，而且有地域和文化差异，必将是一个长期演进的过程。

在业务量、收入等方面与传统话音业务的比较正在取得优势。另一方面，随着智能终端的快速发展，手机已不再像原来那样仅仅提供单纯的话音和短信服务；现在手机同时也是在线支付的购物工具、在线阅读的电子书，导航、微博、SNS、IM等Web2.0应用更多地出现在手机上。据统计，智能手机的应用中，话音应用所占的使用时间仅仅为27%，包括地图、社交网站、电子邮件等Web应用，占据了72%之多。手机不再仅仅是简单的通话工具，大量互联网应用已使得话音业务的份量逐渐降低。

应用为王、内容为王所导致的需求为王，使得我们必须要以提供服务来切入。信息的服务、内容的服务，导致了应用软件就是服务的思想。因为应用软件是服务，才导致了支撑应用软件的开发平台也是服务；因为开发平台是服务，才导致了支撑开发平台运转的基础设施也是服务，产业链的走向倒过来了。在商业运营模式上，话音业务的大幅缩水，移动互联网应用的多样化，智能手机和平板电脑的普及，迫使我们必须重视产业链倒置的事实及其带来的巨大变化。越来越丰富的应用正在大量消耗带宽，电信运营商既没有从通道中获得合理的投资回报，也没有在高附加值的信息服务领域创造更大的企业价值。因此，传统电信行业要改变、要转型。

要充分利用细分市场所形成的各种各样的服务，产业链上其他各方对信息基础设施服务的依赖性，获取传统电信行业的依附优势。电信运营商的依附优势，一方面，体现在电信网对用户先入为主、偏好依附所形成的自然垄断；另一方面，电信运营商拥有丰富的客户信息等基础数据资源。电信运营商在为用户提供服务的过程中获得了客户的特征信息，包括用户的身份信息、网络使用状态信息、业务使用记录信息等。通过这些信息的挖掘，可以从服务“小众”中抽取更为精细的大众的公共需求，适应信息服务灵活分众、融合资源的需求。

总之，要想不纠结，就要找出路。一是电信行业的金字塔式层次结构要向扁平的组织结构转型，融合资源、弹性配置；二是要发挥信息基础设施的依附优势，从小众中获取大众基本信息，挖掘需求，灵活分众，深度服务；三是要适应信息即服务、软件即服务、平台即服务和基础设施即服务的变化，拓展应用，合作共赢。

云计算实践的试点

2010年10月，工业和信息化部、国家发改委等联合确定了在北京、上海、深圳、杭州、无锡五个城市先行开展云计算服务创新发展的试点示范。应该说，北京的祥云计划、上海的云海计划、无锡的物联网和云计算基地等都已经开始了务实的推动，在全国起到了引领、示范的作用。

基于互联网的云计算，节点可大可小，距离不再重要，因此在五个城市试点的基础之上，我个人认为更应该加强行业的应用和领域的应用示范。如果我们让这五个城市连成线、

形成片、形成特定业务领域的专业服务，示范作用就更强了。

这里我想强调一下基于位置的服务LBS（Location Based Service），如果从北京到深圳可以实现整个区域范围内的导航服务和定位服务，那么这就为物联网的应用提供了基础。大家知道，地理信息系统是每一个行业都需要的基本信息，有了地理信息服务，有了导航定位的精度保证，比方说，导航的精度可以到几十厘米，而不是几米，不仅可以导航到道路，而且要导航到车道，那么就可以实现从北京到上海，乃至到深圳的网络导航。我们还可以有基于地理信息的旅游服务、基于地理信息的教育服务、基于地理信息的社会保险服务、基于地理信息的应急救生服务等。如果每个老百姓、每个农民工的身体健康指标都能够在线测试，在家门口、在打工场就可以得到健康和医疗的服务支持，那会是一个什么情景？民生问题是一个国家的基本问题，如果我们能够通过网络实现民生当中最基本的社会保障，实现物联网的话，那又会是一个什么情景？

当前，物联网正在全球兴起，在人人联网的基础之上，实现物物联网。我想从我个人的科研经历讲一个例子。我曾经做过一个三级倒立摆的控制研究，通过计算机，让三级摆表现出各种不同的动平衡模式，是一个典型的控制问题，要解决多变量、不确定、非线性等难题。在今天这个网络环境下，我们能不能通过互联网在远程实现对这样一个倒立摆的控制呢？比如，能不能在上海通过一个手机来控制北京的一个倒立摆，摆出各种动平衡姿态，如果我们能够做到这一点的话，将来就可以做到通过网络控制空中的飞行器、家庭监护机器人以及电力调度等。因此，我认为物联网的最高境界是机器人联网（Internet of Robots），就是把机器、生产线等通过机器人或智能代理与网络连接起来，实现网络化控制，让云计算真正成为物联网的基石。也许有一天，我们可以通过智能控制卡随时将手机变换成为虚拟遥控器或虚拟卡（身份卡、信用卡、驾驶证、门卡等）。甚至可以想象，设计一个生物芯片植入到人的大脑中实现联网。想得到也许就能看得到，当然这是后话。

总之，我们正进入利用信息和信息技术精确调控物质和能量的时代。云计算的实践，我们要从行业和领域入手，从老百姓得到实惠入手，从解决民生入手，不在于拼服务器集群的规模，不是说五万台服务器的集装箱就伟大，五千台服务器的就不伟大，也不是说把一百万台服务器集中起来就叫云计算。

随着试点的推开，云计算的实践也许还要考虑整个中国IT行业的战略布局，将来是不是要在某些地方设立一些国家级的海量基础数据中心和服务中心，是考虑在电力供应相对丰富的低温地区建立海量数据中心，还是考虑在靠近数据中心最终用户的地区建立呢？大家知道，电力传输是比较昂贵的，而宽带传输消耗的经费是比较少的。中国联通研究院就光纤传输与电能传输的成本和能耗做过一个对比分析，光纤传输的百公里建设成本为210万元，而电能传输的百公里建设成本高达2亿4000万元，后者是前者的一千多倍；电力损耗为5.2千瓦时，电能传输则为25千瓦时，后者是前者的5倍。显然，可以在靠近电厂的低温地区建设大规模的“超级数据中心”，利用宽带通信网络向华东、华南传输数据，以替代大量煤炭和电力向华东、华南的输送，变“以电力网输送电能”为“以电信网输送数据”，以减少二氧化碳的排放，改善生态环境。

结束语

云计算是网络环境下计算资源的优化，是技术的创新，也是商业模式的进步。云计算的实践没有现成的成熟模式可供借鉴，而且有地域和文化差异，必将是一个长期演进的过程。但我相信，在国家有关部委云计算示范试点政策的推动下，有产、学、研、用同仁的共同努力，云计算技术创新和应用实践之路会越走越宽广。

（本文根据李德毅院士在第三届中国云计算大会上的演讲整理而成，感谢马于涛、刘玉超和张海粟博士对本文的支持。）

责任编辑：董世晓（dongsx@csdn.net）

HBase性能深度分析

文 / 刘星

HBase作为BigTable的一个开源实现，随着其应用的普及，用户对它的性能数据愈发关注。本文将为您揭开HBase性能测试的一角，邀您一起参与到对云计算模块性能调优的深度思考中。

对于BigTable类型的分布式数据库应用来说，用户往往会对其性能状况有极大的兴趣，这其中又对实时数据插入性能更为关注。HBase作为BigTable的一个实现，在这方面的性能会如何呢？这就需要通过测试数据来说话了。

数据插入性能测试的设计场景是这样的：取随机值的Rowkey长度为2000字节，固定值的Value长度为4000字节，由于单行Row插入速度太快，系统统计精度不够，所以将插入500行Row做一次耗时统计。

这里要对HBase的特点做个说明，首先是Rowkey值为何取随机数，这是因为HBase是对Rowkey进行排序的，随机Rowkey将被分配到不同的region上，这样才能发挥出分布式数据库的性能优点。而Value对于HBase来说不会进行任何解析，其数据是否变化，对性能是不应该有任何影响的。同时为了简单起见，所有的数据都将只插入到一个表格的同一个Column中。

初次分析

在测试之初，需要对集群进行调优，关闭可能大量耗费内存、带宽以及CPU的服务（例如Apache的HTTP服务），保持集群的宁静度。此外，为了保证测试不受干扰，HBase的集群系统需要被独立，以保证不与HDFS所在的Hadoop集群有所交叉。

实验

那么做好一切准备之后，就开始进行数据灌入，客户端从Zookeeper上查询到Regionserver的地址后，开始源源不断地向HBase的Regionserver上喂入Row。

这里，我写了一个通过JFreeChart来实时生成图片的程序，每3分钟，喂数据的客户端会将获取到的耗时统计打印在一张十字坐标图中，这些图又被保存在制定的Web站点中，并通过HTTP服务展示出来。在通过长时间不间断的测试后，我得到了图1。

图1　插入Row的性能测试

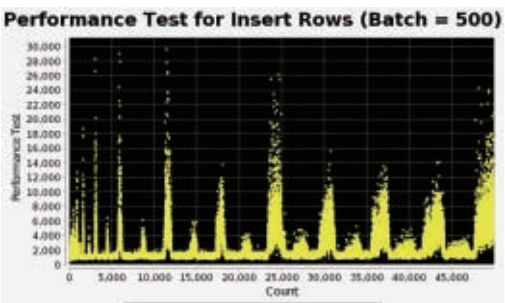


图1好似一条直线上，每隔一段时间就会泛起一个波浪，且两个高峰之间必有一个较矮的波浪。高峰的间隔呈现出越来越大的趋势，而较矮的波浪恰好处于两高峰的中间位置。

解读

为了解释，我对HDFS上HBase所在的主目录下文件，以及被插入表格的region情况进行了实时监控，以期发现这些波浪上发生了什么事情。

回溯到客户端喂入数据的开始阶段，创建表格，在HDFS上便被创建了一个与表格同名的目录，该目录下将出现第一个region，region中会以family名创建一个目录，这个目录下才存在记录具体数据的文件。同时在该表表名目录下，还会生成一个“compaction.dir”目录，该目录将在family名目录下region文件超过指定数目时用于合并region。

当第一个region目录出现时，内存中最初被

写入的数据将被保存到该文件中，这个间隔是由选项“hbase.hregion.memstore.flush.size”决定的，默认是64MB，该region所在的Regionserver的内存中一旦有超过64MB的数据时，就将被写入到region文件中。这个文件将不断增殖，直到超过由“hbase.hregion.max.filesize”决定的文件大小（默认是256MB，此时加上内存刷入的数据，实际最大可能到256MB+64MB）时，该region将被执行split，立即被一切为二，其过程是在该目录下创建一个名为“.splits”的目录作为标记，然后由Regionserver将文件信息读取进来，分别写入到两个新的region目录中，最后再将老的region删除。这里的标记目录“.splits”可以避免在split过程中发生其他操作，起到类似于多线程安全的锁功能。在新的region中，从老的region中切分出的数据独立为一个文件并不再接收新的数据（该文件大小超过了64MB，最大可达到（256+64）/2=160MB），内存中新的数据将被保存到一个重新创建的文件中，该文件大小将为64MB。内存每刷新一次，region所在的目录下就将增加一个64MB的文件，直到总文件数超过由“hbase.hstore.compactionThreshold”指定的数量时（默认为3），compaction过程就将被触发了。在上述值为3时，此时该region目录下，实际文件数只有两个，还有额外的一个正处于内存中将要被刷入到磁盘的过程中。Compaction过程是HBase的一个大动作。HBase不仅要将这些文件转移到“compaction.dir”目录进行压缩，而且在压缩后的文件超过256MB时，还必须立即进行split动作。这一系列行为在HDFS上可谓是翻山倒海，影响颇大。待Compaction结束之后，后续的split依然会持续一小段时间，直到所有的region都被切割分配完毕，HBase才会恢复平静并等待下一次数据从内存写入到HDFS。

理解了上述过程，就必然会对HBase的数据插入性能为何是图1所示的曲线的原因一目了然。与X轴几乎平行的直线，表明数据正在被写入HBase的Regionserver所在机器的内存中。而较低的波峰意味着Regionserver正在将内存写入到HDFS上，较高的波峰意味着Regionserver不仅正在将内存刷入到HDFS，而且还在执行Compaction和Split两种操作。如果调整“hbase.hstore.compactionThreshold”的值为一个较大的数量，例如改成5，可以预见，在每两个高峰之间必然会等间隔地出现三次较低的波峰，并可预见到，高峰的高度将远超过上述值为3时的高峰高度（因为Compaction的工作更为艰巨）。由于region数量由少到多，而我们插入的Row的Rowkey是随机的，因此每一个region中的数据都会均匀的增加，同一段时间插入的数据将被分布到越来越多的region上，因此波峰之间的间隔时间也将会越来越长。

再次理解上述论述，我们可以推断出HBase的数据插入性能实际上应该被分为三种情况：直线状态、低峰状态和高峰状态。在这三种情况下得到的性能数据才是最终HBase数据插入性能的真实描述。那么提供给用户的数据该是采取哪一个呢？我认为直线状态由于所占时间会较长，尤其在用户写入数据的速度也许并不是那么快的情况下，所以这个状态下得到的性能数据结果更应该提供给用户。

再度分析

前面的HBase性能深度分析，提出了一个猜想，是关于调整“hbase.hstore.compactionThreshold”值的假设。猜想的内容为：如果改变该值，例如调整为5，那么耗时图形会在每两个高峰之间出现等间隔的三次较低的波峰，并且高峰将会更加突出，超过上述值较低时的波峰高度。

为了证明这个猜想，我将“hbase.hstore.compactionThreshold”值调整为5，并重新做了数据插入测试，一段时间后，得到如图2所示的性能图形（Y轴表示耗时，X轴为插入次数，Sandy建议这里的Y轴应该改为插入速度，但是

图2　再次的数据插入测试

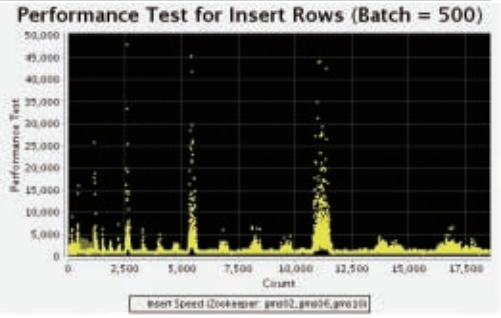


图3　图1的数据分布图

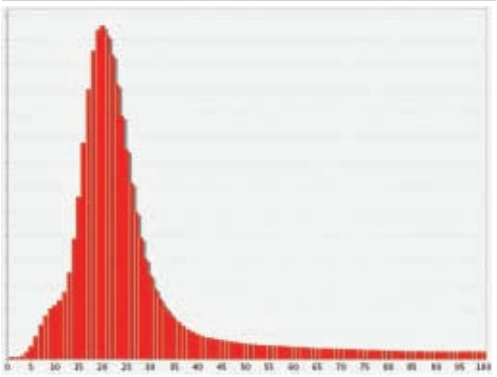
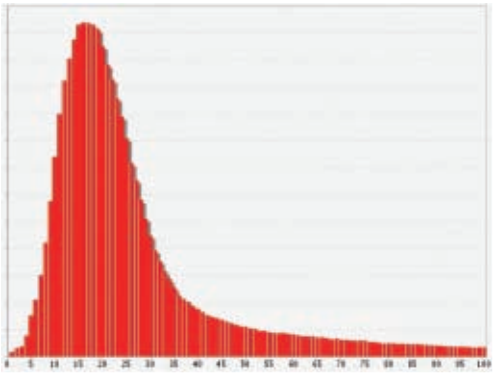


图4　图2的数据分布图



由于前次已经使用了耗时为Y轴，因此改变Y轴显示的工作只能放到下次测试中了）。

通过相比发现，图1和图2的Y轴比例尺是不同的，图1中Y轴最大为30秒，图2中最大为50秒。可见假设中声称低峰会在两个高峰之间等间隔的出现3次的现象的确是成立的。当高峰出现第5次以后，可以从图2看到代表耗时的点的高峰段已经达到了25秒以上，而对于前次来说，高峰段基本上处于20秒左右，由此可以认为Compaction的压力的确是增加了。现在换一个角度来分析这一情况。我为图1制作了一张数据分布图（图3），与图2进行比较（图4）。

虽说第二次测试经历的时间不如第一次，但是基于统计学的观点，分布图的外形是不会受样本容量大小影响的，因此图3和图4可以进行外观上的比较。这两张分布图都是典型的正态分布图，但又不是标准正态分布，原因在于，波峰段的数据影响了正态分布的标准性，表现之一在于右侧的长尾，表现之二在于众数所在位置右移，以至于左侧凸显了一个小波峰。

计算本次分布图与前次分布图中位于右侧长尾部分的数据的标准差（计算公式：$S=\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})/n}$），我们可以得到前次右侧标准差为4.10390692438，而本次右侧标准差为7.12068861446，说明高峰段影响的数据右偏更为严重了。

从外观上表现在右侧长尾在整图比例尺中的宽度和高度要大于前次分布图中的右侧长尾。这说明Compaction的压力增大了。

推导到这里，我发现右侧标准差与Compaction的压力之间是存在显著关系的，今后对Compaction压力增减的估算，貌似可以转换为对右侧标准差的计算。压力增加的比率是否等于标准差的比值呢？这里先做一个标记，等后面有时间再仔细思考一下这个问题。现在假设中的说法“高峰将会更加突出，超过上述值较低时的波峰高度”，应该算是被证明了。

以上论证结束之后，按照惯例还是要提出一些假设和推断。

HBase在已经发布的0.90.x版对Compaction和Split机制作了调整，将Split过程提到了Compaction之前，也就是说，当region目录下，HFiles数目超过“hbase.hstore.compactionThreshold”指定值之后，Regionserver会首先计算一下Compaction之后的文件大小是否会超过“hbase.hregion.max.filesize”确定的Split上限大小，如果超过了，那么HFiles首先被切分，然后才会将切分好的文件转移到新的region中Compaction。这样将大大减小Compaction的压力，由此可以推断，HBase的性能调优必然与“hbase.hstore.compactionThreshold”和“hbase.hregion.max.filesize”这两个值的大小息息相关。理论上，可以将某次设定了确定值的实验中获得的数据代入到一个特定公式中，上述两值作为该公式的自变量，其应变量，即性能数据，将可以轻松地计算出来。

是否真的如此，且待进一步的详细测试。



刘星

趋势科技中国研发中心资深QA，在六年职业生涯中，开发和测试的经验恰好参半。2009年加入趋势科技至今，一直从事基于HBase的分布式存储数据库的研发。

责任编辑：董世晓（dongsx@csdn.net）

云时代的企业应用数据挖掘

文 / 顾茜 赵鹏

本文主要分析了企业面对云时代的SaaS服务时，如何有效地对应用数据进行数据挖掘。首先分析了企业面对海量增长的数据时数据挖掘面临的挑战，其次提出了一种适合云应用环境的数据挖掘模式，最后对该方法进行了总结。

随着云时代的到来和SaaS概念的引入，越来越多的企业开始选择由SaaS应用提供商、运营商等通过互联网平台提供SaaS应用服务，SaaS应用的数据量面临着TB级的增长速度；不同的SaaS应用体系，提供的数据结构也不完全相同，数据有文本、图形甚至小型数据库；SaaS应用数据随着云服务平台的分布性特点，有可能分布在不同的服务器上，如何对这些异构异源的数据进行数据挖掘，是云时代的企业面临的难题。

如何从海量应用挖掘出合理的数据

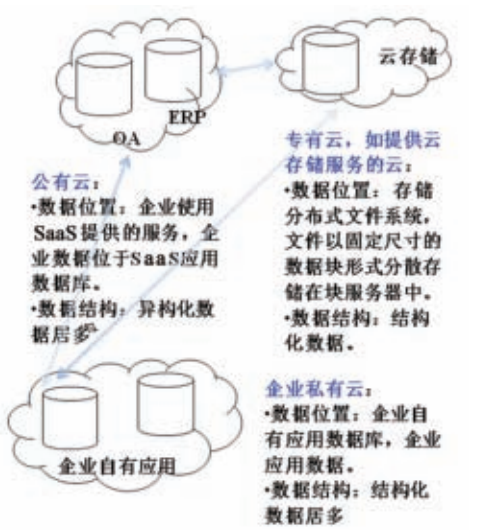
对于企业而言，如何将各种SaaS应用数据进行整合挖掘，提炼出适合其使用的商业信息是企业的一大急迫需求。传统的BI模式大多基于数据仓库，是关系型数据库的模式。面对急剧增长的异构数据，传统的数据仓库和原有的并行计算技术由于挖掘效率低，已经不能解决海量数据挖掘工作，影响着数据的及时提取。

云时代企业数据挖掘也面临如下挑战。

■ 挖掘效率：进入云计算时代后，BI的思路发生了转换。以前是基于封闭的企业数据进行挖掘，而面对引入互联网应用后海量的异构数据（据预计到2020年，爆发式增长的数据量将突破35ZB（1ZB=10亿TB））时，目前并行挖掘算法的效率很低。

■ 多源数据：引入云计算后，企业数据的位置有可能在提供公有云服务的平台上，也可能在企业自建的私有云上，如何面对不同的数据源进行挖掘也是一个挑战。 如图1所示。

图1　企业面临着在不同数据源上进行数据挖掘的挑战



■ 异构数据：Web数据的最大特点就是半结构化，如文档、报表、网页、声音、图像、视频等，而云计算带来了大量的基于互联网模式提供的SaaS应用，如何梳理有效数据是一个挑战。

SaaS应用的数据挖掘希望能够通过海量数据存储平台，引入快速并行的挖掘算法，提高数据挖掘的质量。

适合云应用数据挖掘的模式建议

数据仓库建模阶段

为了应对SaaS应用大量异构数据，引入

XML标记和交换数据。由于XML能够使不同来源的结构化数据很容易地结合在一起，因而使搜索多样的不兼容的数据库成为可能，从而为解决Web数据挖掘难题带来了希望。XML的扩展性和灵活性允许XML描述不同种类应用软件中的数据，从而能描述搜集的Web页中的数据记录。

引入MapReduce算法，提高数据抽取转换的效率。MapReduce算法是Google提出的一个软件架构，用于大规模数据集（大于1TB）的并行运算。当前的实现方法是指定一个Map（映射）函数用来把一组键值对映射成一组新的键值对，指定并发的Reduce（化简）函数用来保证所有映射的键值对中的每一个共享相同的键组。

MapReduce更适合如下场景。

- ETL（数据提取转化加载）类的应用：从多个不同的源读取日志信息；分析以及清理日志数据；执行复杂的变换，比如“会话转换”；决定存储什么样的属性以及把信息装载到DBMS或者其他存储引擎中。
- 复杂分析应用：这种挖掘类型的应用需要对数据进行多步骤的计算和处理，通常一个程序的输出会是另外一个程序的输入，因此很难用单个SQL语句来表示，这种应用场合下，MapReduce是很好的候选方案。
- 半结构化数据：因为不需要对数据的存储进行格式定义，所以MapReduce比较适合处理半结构化数据，这些数据通常都是一些键值对。这些场合下，MapReduce非常适合做ETL的事情。
- 快速实施的系统：完善和健壮的低成本开源解决方案是MapReduce最大的优点。

引入HDFS的分布式存储模式。HDFS系统简单，利于提高实施效率，适合海量数据挖掘。HDFS架构基于GFS体系架构（Google File System，简称GFS，是由Google设计并实现的一个分布式文件系统，基于大量安装有Linux操作系统的普通PC构成的集群系统），但比GFS架构精简。GFS和HDFS都采用“单一主控机+多台工作机”的模式，通过数据分块和复制（多副本，一般是3）来提供更高的可靠性和性能。GFS允许文件被多次或者多个客户端同时打开以追加数据，以记录为单位。而在HDFS中，文件只允许一次打开并追加数据。GFS中采用主从模式备份Master的系统元数据，当主Master失效时，可以通过分布式选举备机接替主Master继续对外提供服务，而由于Replication及主备切换本身有一定的复杂性，HDFS Master的持久化数据只写入到本机（可能写入多份存放到Master机器的多个磁盘中防止某个磁盘损害），出现故障时需要人工介入。GFS通过内部采用Copy-on-Write的数据结构实现集群快照功能，而HDFS不提供快照功能。

引入Hive架构。Hive是建立在Hadoop上的数据仓库基础构架，是一种可以存储、查询和分析Hadoop中大规模数据的机制，提供了一系列工具用来进行数据ETL操作。Hive定义了简单的类SQL查询语言，称为HQL，它允许熟悉SQL的用户查询数据。这个语言也允许熟悉MapReduce开发者自定义Mapper和Reducer来处理内建的Mapper 和Reducer无法完成的复杂分析工作。

数据挖掘阶段

引入数据分析中间件，提供数据处理、数据探索、数据建模及模型应用等一系列功能，开发多种数据挖掘算法和统计建模方法，并能够方便、快速、高效地处理海量数据，为商业智能的应用提供更方便、更灵活的工具和服务。

数据呈现阶段

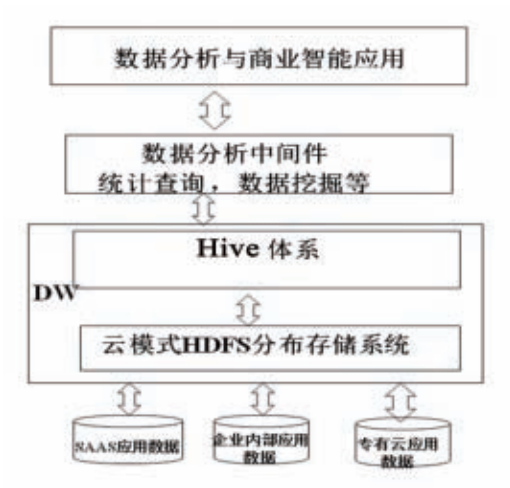
BI作为云计算的一种SaaS服务提供给企业，建立行业数据库。面对林林总总的SaaS应用，BI同样可作为一种SaaS服务提供给企业。同时，数据挖掘工具进行数据分析，可以发现重要的数据模式，这对构建知识库做出了巨大贡献——数据和信息之间的鸿沟要求系统地开发数据挖掘工具，将数据“坟墓”转换成知识“金块”。

数据挖掘模式图

我们设计的数据挖掘模式图如图2所示。

- 企业数据层：企业数据来源于各类应用，如 SaaS应用、企业内部应用数据和专有云

图2　数据挖掘模式图



应用。

- 数据仓库层：主要引入HDFS分布存储系统和Hive体系架构，通过MapReduce算法对数据梳理和提取。
- 数据挖掘层：引入基于XML数据分析中间件，实现统计查询和数据挖掘功能。
- 数据分析与BI应用层：将BI以SaaS服务的模式提供给企业使用。

总结

随着云时代的到来，企业面临的应用方式更加多元化，通过云的手段提供海量数据挖掘的方法，提高了挖掘的效率，增加了挖掘的精度，更利于挖掘应用的推广以及专业的行业知识库的构建。



顾茜

中国电信股份有限公司北京研究院工程师，目前从事云计算领域的研究工作。



赵鹏

中国电信股份有限公司北京研究院高级工程师，主要研究方向为信息系统规划与设计、语义网络和知识工程。

责任编辑：董世晓（dongsx@csdn.net）

中科院IT工程硕士研究生
2011年招生信息

以“精技术、懂管理、会做人”为目标，专注于IT领域高层次复合型人才培养；以创新型的“研究生培养与IT人职业发展”相结合的教育体系，为学生发展注入动力之源。

中科院2011年招生领域：

- 软件工程
- 电子与通信工程
- 计算机技术
- 控制工程
- 集成电路工程
- **软件科技理学硕士**：中科院与香港理工大学联合培养，获得香港理工大学硕士学位证书，可获得国际通用GPA记录，留学申请更有竞争力！

招生对象：
大学本科毕业的IT在职人员或应届毕业生，软件科技理学硕士可招收具有三年软件领域工作经验的专科生。

学习方式：
在职学习（周六、日上课）促进职业提升。

报名时间：
2011年5月至7月中旬



咨询详情：
中国科学院研究生院计算与通信工程学院
联系电话：010-88256568/6547/9429
电子邮件：ccce@gucas.ac.cn
学院网站：Http://ccce.gucas.ac.cn
联系地址：中国北京玉泉路19号甲
邮政编码：100049



架构师接龙

李 祎 VS. 吴朱华

主持人：冯大辉

现任丁香园 （http://www.dxy.cn）网站CTO。曾历任支付宝架构师、数据库团队负责人等职。

李祎： 你的网站是否用到类似于MapReduce的框架，例如Hadoop？用来解决哪些问题？使用效果如何？

吴朱华： 与过去栏目嘉宾主要是负责应用系统设计的架构师不同，我主要侧重于底层核心系统的开发，所以对于李祎先生提出的一些问题，会尽可能地从我自身的角度出发，给出一个不同的视角。

关于MapReduce，在我接触的朋友和客户当中，有一些人已经采用了Hadoop，主要都是用于解决海量数据处理的问题。从我的角度而言，Hadoop优点在于其灵活性、简单性和并行性：首先，它不限定输入数据的格式（即无论是类似PDF和文本的非结构化数据，还是类似数据库表的结构化数据），并且其能支持比SQL语言更多样的运算；其次，Hadoop的逻辑也较简单，就是Map、Shuffle和Reduce这三个阶段；还有，就是并行性，Hadoop能够将多个工作拆分成许许多多的工作包，并分给多达几千台物理机，通过这样方式来最大化I/O，从而加快整体的处理速度。

虽然Hadoop有上面提到这些优点，但在使用方面，我有两点顾虑，其一，我个人认为用Hadoop来分析结构化数据不是特别合适，因为就像数据库大师Michael Stonebraker在《MapReduce：一个巨大的倒退》一文提到的那样，通过几十年的时间，数据库在压缩、索引和调度等方面已经积累了很多优秀的技术，如果把这些技术利用好的话，应该更好和更快地处理海量结构化数据，而且更方便，因为无需将数据在数据库和外部Hadoop集群之间导进和导出；其二是在成本和实用性方面，小型或者单机的Hadoop集群在和传统Grep等小工具相比，根据我的理解无论在性能还是在使用等方面都没有明显的优势。

Hadoop的确有很多优点，特别是对于海量的非结构化数据，Hadoop几乎是唯一可行的方案，但我个人还是倾向于尽可能地将非结构化数据转化为结构化数据，因为无论在存储方面还是分析方面，都有更多现成的工具，比如，分布式数据库，有些分布式数据库能通过压缩将数据的存储空间缩小为原先的5倍以上等，这样不仅能降低存储成本，而且可以加快数据分析的速度。



提问嘉宾

李 祎

卓望信息技术公司系统架构师，主持开发了139网站的用户平台、社区平台、监控平台。现负责海量数据存储访问、推荐引擎架构和图形数据库开发。曾任西门子手机技术创新部高级工程师。关注移动和互联网领域的新技术。



回答嘉宾

吴朱华

《云计算核心技术剖析》一书作者，曾在IBM中国研究院参与多款云计算产品的研发，现为上海人云科技创始人兼总经理，海量数据分析数据库YunTable和PaaS系统YunEngine核心架构师，对云计算和NoSQL等有深入研究和实践。

说到底，其实Hadoop和传统的数据库并没冲突，如何更好地组合和融合才是关键，无论是结构化数据，还是非结构化数据，都存在着很多机会。

李祎： 前段时间亚马逊的云服务EC2因机房断电而瘫痪，你对亚马逊的这次灾难和云计算有何看法?

吴朱华： 身为一个专注于云计算并且长期从事于此的人而言，谈到这里的时候，不得不让我联想到两件事情，首先是云计算与电力的关系，尼古拉斯•卡尔所著的《大转变》一书中描述了整个IT产业正在经历一个类似电力从发电机发电到电厂供电的巨大转变，虽然拿电厂和云计算中心相比并不是毫无偏差，但将这两者之间的进行类比我觉得还是很合适的；其次，在我小的时候，家里面经常停电，就像当年我们总是不断地重装系统那样，但现在家庭用电已经非常稳定了，停电的情况已经越来越少，同时现在系统也很少被重装。我说上面两件事情的原因是，我觉得云计算是IT技术一个非常自然的更替，就像过去每个公司都有发电机一样，随着时间的发展，由于在成本和使用方面的优势，云计算就会像电厂那样最终成为主流，当然在这个过程中，肯定会出现阵痛，比如亚马逊的云服务最近出现的宕机事件等。但经过5~10年的发展，我相信云计算相关的服务和技术将会更稳定。

云计算并不怎么神秘，就是下一代的IT技术，所以我不奇怪有很多厂商跳出来说他们做的就是云计算，当然Google的云计算数据中心和相关的分布式技术是其中最鲜明的代表。

李祎： 现在业界有一股将应用去SQL化的风潮，你从哪些方面来考虑一个可扩展的网站和大型应用是使用NoSQL还是SQL-based?

吴朱华： 这个问题在我实际工作中碰到过，我个人感觉关系型数据库有点“Over-Engineering”，SQL承载功能较多，而且还有很多私有功能，使得在这方面的创新和改进特别艰难，尤其是在伸缩性这块，所以去掉一些SQL，在云计算时代还是非常有益处的，特别是当处理海量数据时。谈到使用NoSQL还是SQL-based，我个人感觉在很长一段时间里，我们将很难抛离SQL这套规范，因为它不仅经典和实用，而且是每个程序员的必修课。我比较倾向在SQL基本集和SQL全集之间选择，在这个基础上，由于支持SQL全集，并且支持可扩展的的解决方案，其价格是比较天文数字的，对于普通的互联网公司而言可望而不可即，所以我觉得支持SQL基本集的分布式数据库更合适。

李祎： 你是如何理解KISS（Keep it Simple and Stupid）的？在实际项目中如何做到?

吴朱华： 关于KISS原则，大家已经聊了很多关于设计方面的想法，的确KISS这个原则言简意赅，而且非常可行。但我也觉得KISS这个原则，并不仅适用于设计，在我们平时的编程过程中也非常适用，最简单的体现是就是“不要写重复代码”，可以理解为“不要复制和粘贴”。根据多年的编程经历，我感觉这个规则非常重要，如果为了短期的方便，而大量地复制和粘贴代码，将会造成在系统很多模块中出现冗余代码，这不仅极大地增加了系统代码量，而且严重影响代码的可读性和可维护性，这种痛苦只有那些有实际经历的人才能体会。我个人感觉不仅需要在设计上强调KISS，而且应该在实际编程时尽量避免重复代码的出现。

李祎： 当计划创建一个新系统时，你从哪些方面考虑它的可伸缩性?

吴朱华： 我曾经和一位客户谈到这个问题，我给了他两个建议。首先，确定伸缩的瓶颈，由于负载均衡技术的发展，再加上Web端本身基本上是无状态的，所以Web层的伸缩性不应该是重点考量的。但是由于数据库本身有状态，再加上其他一些因素，使得数据持久层的伸缩成为关键，所以在大多数情况下，数据库那端是最主要考虑的。其次，在数据持久化方面，要考虑一下工作负载的类型，是以OLAP为主，还是OLTP为主，大致读写的数据量级，以及数据量是不是可估的，之后再基于这些因素来判断具体用于数据持久化的数据库产品。

责任编辑：董世晓（dongsx@csdn.net）



主持人：张银奎
《软件调试》一书作者，从事软件开发和研究10余年，对IA-32架构、操作系统内核、虚拟技术，尤其对软件调试有较深入的研究。翻译（合译）作品包括《数据挖掘原理》、《机器学习》、《人工智能：复杂问题求解的结构和策略》、《观止——微软创建NT和未来的夺命狂奔》等。

经典阅读器的经典死锁

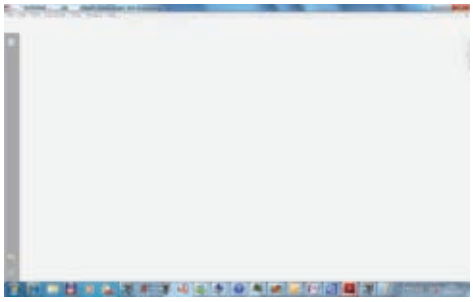
作者调侃“经典的软件，发生的死锁也很经典”，详细描述了著名的PDF阅读器存在的一个死锁问题。

端午节三天假期，有两整天下雨，还有一天阴天，好像随时都可能下起来。这样的假期非常适合呆在家里看书。拿一本书捧在手中，泡一杯茶置于案侧。闻茶香悠悠，听雨声潺潺，真是惬意。当然，除了读书，这个假期还有一个任务，就是写这个专栏的文章。这一期写什么呢？Windows 7、MSN Messager、PowerPoint、Onenote？微软的软件写了很多了，以至于有一次见到一位朋友，他笑着问我，为什么不写写他们公司的软件？我说，你们公司的软件质量高，没有问题啊，大家都笑了。这当然是开玩笑，哪里能找到没有问题的软件呢？问题俯拾即是，当然要有所选择，比如须是比较著名的软件，须出自比较有名的公司，须是比较复杂而且有代表性的问题……本期选什么呢？既然以阅读开头，那么就写著名的PDF阅读器的一个死锁问题吧。经典的软件，发生的死锁也很经典。

迟迟打不开的PDF附件

故事得从一封邮件讲起。有一天晚上，当我检查GAMIL邮箱时，有一封邮件让我眼前一亮，打开邮件，正文很短，关键的内容都在附件中。附件是一个PDF文件，选择附件后，GMAIL做了一番扫描，没有发现病毒，显示出View和Download两个链接。View功能很少工作，于是选择Download。接下来是包含Open、Save和Cancel按钮的文件下载对话框，选择Open直接打开。几秒钟后，熟悉的PDF阅读器窗口按照预期出现了，但是期望看到的文件内容却迟迟没有显示出来，窗口的用户区一片空白，如图1所示的样子（标题栏中的文件名涂去若干个字）。

图1 迟迟打不开的PDF附件



等了一阵子后，窗口终于有所变化，但文件内容还是没有显示出来，而是窗口标题中增加了一对括弧，括弧里面写着“Not Responding”。如果直译为中文，或许可以译为“正在不响应”。根据多年使用Windows操作系统的经验，程序挂住了。确切地说，是Windows子系统发现PDF阅读器长时间不处理窗口消息，在其标题条中打上“不响应”标签，目的是提示用户，对于这样的程序，点鼠标，敲键盘都没有什么用了，除非是强制关闭。

关掉重来否？

对于这种情况，最快的解决方法是把挂死的程序实例杀掉，然后再重新打开附件，也就是关掉重来。但我却舍不得这么做。因为这么做，很可能就错过了一个“千载难逢”的Bug，如果以后再想复现它，也许要花很多很多的时间。因为这样的考虑，我没有急于关掉PDF阅读器，而是遵循我的职业习惯，启动调试器，附加到这个进程，执行以下命令产生了一个转储文件：

```
.dump /mfh c:\dumps\pdf\acrobat.dmp
```

命令选项中的/m代表Mini Dump，但这个名字并不准确，Windows下的用户态转储文件一般都是使用MiniDumpWriteDump这个API来写的，因此一般都叫Mini Dump，但其实Mini Dump不一定真的Mini，可能很大。比如第二个选项f就代表Mini Dump中要包含所有内存数据，也就是这个进程中可以访问的所有已提交内存页。

Adds full memory data to the minidump. All accessible committed pages owned by the target application will be included.（WinDBG**帮助文件**）

接下来的h选项代表包含句柄信息，这对于调试程序挂死也经常是有价值的。

以上命令执行后，便产生了一个磁盘文件，大小为108MB，不算Mini吧？有了转储文件后，便可以杀掉当前的进程实例，然后再重新打开邮件附件了。待时间充裕时，可以打开转储文件一探挂死的原由，以下内容便是分析转储文件的结果。

“不响应”的三种原因

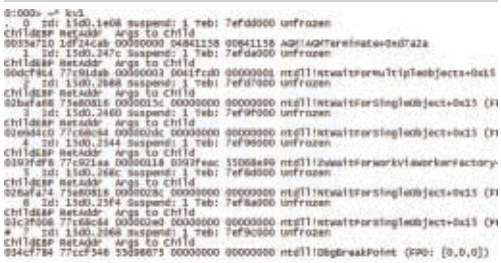
可以把应用程序不响应的原因归纳为如下三类：死锁、死循环和粘滞在内核态。一旦程序的UI线程遭遇以上三种情况之一，便难以脱身去处理窗口消息；无法处理窗口消息就意味着失去响应了。后两种情况我们以前曾介绍过，比如我们曾讲过的PowerPoint例子就是UI线程因为调用SendMessage API广播DDE消息而陷入僵局，MSN的例子是关于死循环的。这一次讲的是死锁的例子。

重新倒好一杯茶，启动WinDBG，File→Open Crash Dump…，打开以前保存的转储文件，让时光倒流，回到当初的那个瞬间，思考如何寻找线索，探索突然挂死之谜。

刚开始学习WinDBG的朋友经常抱怨WinDBG的命令太多，语法古怪，不好记忆。但我却很适应、喜欢，甚至可以说是热爱这套命令，或许是“日久生情”吧。每次遇到问题，总有那么一群命令会涌现到脑海里，“争着吵着”要试一试，比如用~*观察线程概况；用~* k逐个显示每个线程的执行轨迹（栈回溯）；用!peb看进程的概况；用lm观察进程里的模块信息，等等。

图2给出了执行~* kv1命令的结果（为节约篇幅，去掉了空行和行尾部分信息，下同）。可以看到进程中一共有8个线程，其中7号是调试器注入的远程中断线程，不在考虑之列。0号前的.代表是调试会话中的当前线程，7号线程前的“#”代表中断到调试器（产生转储）时的当前线程。

图2 观察线程概况



从栈回溯的最上层函数可以看到，除了远程中断线程外的7个线程中有6个是在执行不同形式的Wait函数。这些函数要么以Nt开始，要么以Zw开始，表明它们都是系统服务在NTDLL.dll中的桩函数（Stub）。这意味着这6个线程都因为调用系统服务而进入到内核态了，而且因为它们调用的都是等待函数，可能已经在内核态被挂起而进入睡眠状态了。

有谁在等待？

如何知道1～7号这六个调用等待函数的线程会等待多久呢？可以尝试分析保存在栈上的函数参数，因为我们没有符号，所以需要根据函数原型来手工解析参数。首先看NtWaitForSingleObject，通过MSDN很容易找到它的函数原型：

```
NTSTATUS WINAPI NtWaitForSingleObject(
  __in  HANDLE Handle,
  __in  BOOLEAN Alertable,
  __in  PLARGE_INTEGER Timeout
);
```

第三个参数是用来指定等待时间的，如果为NULL，则表示永久等待，相当于调用WaitForSingleObject API时指定INFINITE常量。

图3 等待函数及其参数

```
0:005> kv
ChildEBP RetAddr  Args to Child
026afa74 75e80816 0000028c 00000000 00000000 ntdll!NtWaitForSingleObject+0x15 (FPO: [3,0,0])
026afae0 75a11184 0000028c ffffffff 00000000 KERNELBASE!WaitForSingleObjectEx+0x98 (FPO: [Non-F
026afaf8 75a11138 0000028c ffffffff 00000000 kernel32!WaitForSingleObjectExImplementation+0x75
026afb0c 56c537e6 0000028c ffffffff 00000000 kernel32!WaitForSingleObject+0x12 (FPO: [Non-Fpo])
```

```
#define INFINITE            0xFFFFFFFF
// Infinite timeout
```

看一下6号线程的栈回溯可以清楚的看到WaitForSingleObject、WaitForSingleObjectEx与NtWaitForSingleObject的相互关系（图3）。

根据函数原型，再观察图2中的参数数据，可以看到调用NtWaitForSingleObject的四个线程（2、3、5、6）指定的Timeout参数都是0（NULL）。这意味着这4个线程都是无限期等待第一个参数所指定的内核对象，如果没有线程改变这些内核对象的状态，那么这4个线程便会一直等待下去。

再来看下1号和4号线程，它们分别调用了NtWaitForMultipleObjects和ZwWaitForWorkViaWorkerFactory函数，观察它们的栈回溯（图4），可以看出它们都是线程池的工作函数，在等待任务。这样看来这两个线程是在等待事情做，从目前的状态来看，它们不会去唤醒前面那4个永久等待的线程，除非有线程发起了一个工作任务（Work Item），而且这个工作任务在执行时会改变那是个线程所等待的内核对象。谁会创建这样的工作任务呢？希望只能继承在0号线程上了。

图4 线程池工作函数在等待任务

```
0:005> ~1 k
ChildEBP RetAddr
00dcf914 77c91dab ntdll!NtWaitForMultipleObjects+0x15
00dcfae0 75a13677 ntdll!TppWaiterpThread+0x33d
00dcfab4 77c69d72 kernel32!BaseThreadInitThunk+0xe
00dcfaf4 77c69d45 ntdll!__RtlUserThreadStart+0x70
00dcfb0c 00000000 ntdll!_RtlUserThreadStart+0x1b
0:005> ~4 k
ChildEBP RetAddr
0393fdf8 77c921aa ntdll!ZwWaitForWorkViaWorkerFactory+0x12
0393ff50 75a13677 ntdll!TppWorkerThread+0x216
0393ff64 77c69d72 kernel32!BaseThreadInitThunk+0xe
0393ffa4 77c69d45 ntdll!__RtlUserThreadStart+0x70
0393ffbc 00000000 ntdll!_RtlUserThreadStart+0x1b
```

0号线程

如此看来，在所有7个线程中（不包括远程中断线程），有4个因为永久等待它们“心仪”的内核对象而睡觉，有2个是线程池的工作线程，等待任务，无事可做。剩下的0号线程在做什么呢？它是不是正在为唤醒其他线程而努力呢？

使用~0s命令切换到0号线程（图5），从线程上下文来看，这个线程正在执行用户态的代码。

图5 观察0号线程

```
0:005> ~0s
eax=0100036a ebx=0035e78c ecx=01000100 edx=00000000 esi=0493a118 edi=0035e790
eip=5dcae4c2 esp=0035e65c ebp=0035e710 iopl=0         nv up ei ng zr na pe cy
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000283
AGM!AGMTerminate+0xd7a2a:
5dcae4c2 8b0f            mov     ecx,dword ptr [edi]  ds:002b:0035e790=01000100
0:000> u
AGM!AGMTerminate+0xd7a2a:
5dcae4c2 8b0f            mov     ecx,dword ptr [edi]
5dcae4c4 8b13            mov     edx,dword ptr [ebx]
5dcae4c6 3bca            cmp     ecx,edx
5dcae4c8 7ddd            jge     AGM!AGMTerminate+0xd7a0f (5dcae4a7)
```

对于普通的Windows程序，通常我们很难在调试器中看到一个线程在用户态执行，因为用户态的代码通常执行很短的时间，然后便调用系统服务进入到内核态执行了。例外的情况便是用户态的代码有耗时较长的循环。做一下简单的反汇编，可以看到这个线程真的在做循环，正在执行的是将变量值读到寄存器ECX中，而后再把另一个变量读到EDX中，二者比较，大于或者等于时都向上跳继续循环。从图5可以知道，EDI所代表变量的值为0。观察EBX所代表变量的值：

```
0:000> dd @ebx l1
0035e78c  00000000
```

二者相等，会跳上去继续循环。如此分析一番，可以看出0号虽然没有调用等待函数，但在使用循环方式在做忙等待（Busy wait），每次循环时会调用一下pause指令。

```
5dcae4c0 f390            pause
```

看来，0号线程也是在等待，没有任何迹象去唤醒其他线程。因为在这个忙循环中，0号线程没有调用处理窗口消息的函数（GetMessage/PeekMessage等），所以界面没有更新，被窗口子系统打上了“不响应”的标签。

0号线程在等待谁呢？既然是在用户态忙等待，这通常意味着是在等待同一个进程内的其它线程。排除两个线程池的工作线程，0号线程等待的应该是那4个永久等待的线程。这个分析结果与实际情况非常一致，这样等待是没有结果啊，等到地老天荒也没有用。

锁

对于这样的问题，WinDBG专门设计了一系列命令，等待往往与锁有关，因为要进入的区域被加锁了，所以才要等待啊。扩展命令!locks可以用来显示进程中处于锁定状态的所有关键区，图6是执行这个命令的结果。

图6 锁定的关键区

```
0:000> !locks

CritSec CoolType!CTCleanup+ec1dd at 5f15a814
WaiterWoken        No
LockCount          1
RecursionCount     1
OwningThread       2460
EntryCount         0
ContentionCount    1
*** Locked

CritSec CoolType!CTCleanup+ec875 at 5f15aeac
WaiterWoken        No
LockCount          1
RecursionCount     1
OwningThread       25f4
EntryCount         0
ContentionCount    1
*** Locked

CritSec CoolType!CTCleanup+ec9c1 at 5f15aff8
WaiterWoken        No
LockCount          0
RecursionCount     1
OwningThread       25f4
EntryCount         0
ContentionCount    0
*** Locked

Scanned 541 critical sections
```

很多朋友应该都曾用过关键区，它具有使用简单而且开销低这两大优点，在Windows程序中使用得非常普遍。从图6来看，进程中一共有541个关键区，其中三个是处于锁定状态的，通俗点说，有三个关键区内部有线程在里面。

```
EnterCriticalSection(&g_CountCS);
// 访问共享资源
LeaveCriticalSection(&g_CountCS);
```

从实现的角度来讲，每个关键区对应于下面这样一个RTL_CRITICAL_SECTION结构：

```
0:000> dt _RTL_CRITICAL_SECTION
ntdll!_RTL_CRITICAL_SECTION
   +0x000 DebugInfo          : Ptr32 _RTL_
CRITICAL_SECTION_DEBUG
   +0x004 LockCount          : Int4B
   +0x008 RecursionCount     : Int4B
   +0x00c OwningThread       : Ptr32 Void
   +0x010 LockSemaphore      : Ptr32 Void
   +0x014 SpinCount          : Uint4B
```

其中，LockCount字段用来标识关键区的锁状态，RecursionCount字段用来记录递归次数，用来支持同一个线程多次进入关键区，OwningThread字段用来记录进入（拥有）这个关键区的线程ID，LockSemaphore字段用来记录与这个关键去对应的事件对象，当有线程需要等待这个关键区时，便等待这个事件对象，这个事件对象是按需要创建的，如果LockSemaphore字段为空，意味着这个关键区非常“流畅”，从来没有线程在此“堵塞”。

有了这个基础后再来看图5，它包含了关键区的部分字段内容，但没有包含很重要的LockSemaphore字段，好在WinDBG有另一个命令可以弥补这个不足，这就是专门用来观察关键区的!cs -l命令，见图7。

图7 更详细的关键区信息

```
0:006> !cs -l
-----------------------------------------
DebugInfo          = 0x004135e8
Critical section   = 0x5f15a814 (CoolType!CTCleanup+0xEC1DD)
LOCKED
LockCount          = 0x1
WaiterWoken        = No
OwningThread       = 0x00002460
RecursionCount     = 0x1
LockSemaphore      = 0x2E0
SpinCount          = 0x00000000
-----------------------------------------
DebugInfo          = 0x00413638
Critical section   = 0x5f15aeac (CoolType!CTCleanup+0xEC875)
LOCKED
LockCount          = 0x1
WaiterWoken        = No
OwningThread       = 0x000025f4
RecursionCount     = 0x1
LockSemaphore      = 0x2DC
SpinCount          = 0x00000000
-----------------------------------------
DebugInfo          = 0x00413728
Critical section   = 0x5f15aff8 (CoolType!CTCleanup+0xEC9C1)
LOCKED
LockCount          = 0x0
WaiterWoken        = No
OwningThread       = 0x000025f4
RecursionCount     = 0x1
LockSemaphore      = 0x0
SpinCount          = 0x00000001
```

观察图6和图7中的OwningThread信息，三个锁定的关键区属于两个线程，分别是2460和25f4，后者拥有两个。观察图2，6号线程的ID便是25f4，根据前面的分析，它在永久等待参数1指定的内核对象，还是根据图2，参数1的值是2e0，再看一下图7，线程2460所拥有的关键区的事件对象正是2e0，看来，已经拥有两个关键区的25f4线程在等待进入2460所拥有的关键区。再看图2，3号线程的ID是2460，它在等待2DC句柄，这正是线程25f4所拥有的第一个关键区的事件对象，原来2460在拥有了一个关键区后想进入25f4已经拥有的关键区。哇，25f4想进入2460所拥有的，2460想进入25f4说拥有的，二者死锁了，和教科书上介绍的情况一样，经典！原则上讲，如此死锁是因为忽略教科书常有的一句话：如果可能有多个线程需要同时获取多个锁，那么获取的顺序一定要一致。

本期问题

上期答案： 可以在WinDBG中用!erro命令来显示MsiDetermine-PatchSequence函数返回错误码0x6e的含义，即：

```
!error 6e
Error code: (Win32) 0x6e (110) - The system cannot open the device or file specified.
```

目前，MSDN上的函数描述还没有包含这个错误码。

本期问题： 我们在介绍关键区结构的各个字段时，没有介绍SpinCount字段，你知道这个字段的含义和使用方法吗？

编者说明：

- 投稿信箱：contest@csdn.net
- 联系方式写在一个单独的TXT文件里，包括以下几项：
 1）姓名
 2）工作单位或学校
 3）电话联系方式
 4）邮寄地址
 5）E-mail地址
- 解答提交时间，最好早于当月15日。

常见多线程Bug分析

文 / 陈冠诚

在编写多线程程序时，最困难的一个环节恐怕就是调试多线程Bug。本文介绍了常见的两类多线程Bug及其解决方案。

在编写多线程程序时，最困难的一个环节恐怕就是调试多线程Bug了。多线程程序在多核CPU上执行时的交错顺序是不确定的，同一个多线程程序在每次执行时可能都会有不同的交错顺序；与此同时，大多数多线程Bug却只在某个特定的交错执行顺序时才会出现，从而导致多线程Bug难以复现，调试极其困难。现在已经有不少能检测出部分多线程Bug（例如死锁、数据竞跑等）的工具，但是能完美调试多线程Bug的工具仍未出现。在这个前提下，唯有程序员自己对常见的多线程Bug有较深入的理解才能在进行并行编程时有意识地避免和修正多线程Bug。已经有很多文章介绍过死锁、数据竞跑等常见的多线程Bug，但是违反原子性和顺序性这两类多线程Bug却鲜被提及。本文主要的目的就是向大家介绍常见的两类多线程Bug及其解决方案。

多线程程序执行模型

在分析多线程Bug之前，我们先来简单回顾一下多线程程序是怎么执行的。从程序员的角度来看，一个多线程程序可以看成是由每个线程的指令交错在一起执行的，即顺序一致性模型（Sequential Consistency因为多核CPU和编译器的乱序优化，多线程程序在多核CPU上很可能不是按照顺序一致性模型被执行的。在这里为了便于大家理解而使用顺序一致性模型进行阐述）。该模型有两个特征：每个线程内部的指令是按照代码指定的顺序执行的（Program Order），但是线程之间的交错顺序是任意的、不确定的（Non Deterministic）。

有一个形象的例子可以帮助更好地理解顺序一致性模型。用左右手分别代表两个线程，从食指到小拇指的四根手指代表每个线程要依次执行的四条指令。

（1）对每个线程（手）来说，它的四条指令（手指）的执行顺序必须是从第一条指令（食指）依次执行到最后一条指令（小拇指）；

（2）这两个线程（手）的八条指令（八根手指）可以在满足（1）的条件下按任意交错顺序执行（例如可以是左1、左2、右1、右2、右3、左3、左4、右4，也可以是左1、左2、左3、左4、右1、右2、右3、右4等等），并且此执行顺序是全局一致的（即每个线程都遵守此执行顺序）。

现在让我们来看看程序员在编写多线程程序时是如何犯错的。

违反原子性

何谓原子性？简单来说，如果某个操作是不可被分割的，那么该操作就具备原子性。大部分程序员在编写多线程程序员时仍然是按照串行思维来思考，他们习惯性认为一些简单的代码肯定是具备原子性的。

例1：

```
    线程 1
线程 2
S1: if (thd->proc_info)
...
{                                   S3:
thd->proc_info=NULL;
  S2: fputs(thd->proc_info,...)
}
```

这个来自MySQL的多线程Bug的根源就在于，程序员误认为线程1在执行S1时如果从thd->proc_info读到的是一个非空的值的话，在执行S2时thd->proc_info的值肯定也还是非空的，所以可以调用fputs()进行操作。事实上，{S1,S2}这两条语句组合到一起之后并不是原子操作，所以它们可能被线程2的S3打断，即如果按S1→S3→S2的顺序执行的话，线程1运行S2时就会出错（注意，虽然这个Bug是因为多线程程序执行顺序的不确定性造成的，但它违反的是

程序员对这段代码是原子的期望，所以这个Bug属于违反原子性的Bug）。

上面例子的对象是两条语句，所以很容易看出来它们的组合不是原子的。事实上，有些看起来像是原子操作的代码其实也不是原子的。其中最著名的例子莫过于多个线程执行类似“x++”这样的操作了。这条语句本身并不是原子的，因为它在大部分硬件平台上其实是由三条语句实现的：

例2：

```
mov eax,dword ptr [x]
add eax,1
mov dword ptr [x],eax
```

同样，下面例子中的“r.Location = p”也不是原子操作，因为它其实是两个操作组成的：“r.Location.X = p.X”和“r.Location.Y = p.Y”。

例3：

```
struct RoomPoint {
   public int X;
   public int Y;
}
RoomPoint p = new RoomPoint(2,3);
r.Location = p;
```

从根源上来讲，如果你想让这段代码真正按照你的心意来执行，就需要仔细考虑是否会出现违反本意的执行顺序，特别是涉及的变量（例如thd→proc_info）在其他线程中有可能被修改的情况，也就是数据竞争（Data Race）（严格意义上来讲，Data Race只是Atomicity Violation的一个特例，Data Race Free不能保证一定不会出现Atomicity Violation。例如文中Java实现的那个Concurrent Queue的例子，严格意义上来讲它并没有data race，因为isEmpty()和poll()都是线程安全的调用，只不过它们组合起来之后会出现违反程序员本意的Atomicity Violation，所以要用锁保护起来）。如果有两个或更多线程同时对同一个内存地址进行操作，而且它们之中至少有一个是写操作，数据竞争就发生了。

有时候数据竞争可能隐藏得很深，例如下面的Parallel.For看似很正常：

例4：

```
Parallel.For(0, 10000,
    i => {a[i] = new Foo();})
```

实际上，如果我们去看看Foo的实现：

例5：

```
class Foo {
    private static int counter;
    private int unique_id;
    public Foo()
       {
        unique_id = counter++;
       }
}
```

因为Foo类中的counter是静态变量，在多个线程并行执行Parallel.for循环时，多个线程会调用Foo()构造函数并执行counter++，从而就会产生数据竞跑。想避免违反原子性Bug，其实就是要从根本上保证没有数据竞跑（即Data Race Free）。

违反原子性的解决方案

对于违反原子性这一类多线程Bug，常见的解决方案大致有三个（可结合使用）。

（1）把变量隔离起来：只有一个线程可以访问它（isolation）。

（2）把变量的属性定义为immutable的：这样它就是只读的了（immutability）。

（3）同步对这个变量的读写：比如用锁把它锁起来（synchronization）。

例如下面这个例子里面x是只读的；而a[]这个数组则通过index i隔离起来了，即不同线程处理a[]中不同的元素。

例6：

```
Parallel.For(1,1000,
i => {
    a[i] = x;
});
```

下面这个例子在构造函数中给x和y赋值（此时别的线程不能访问它们），保证了隔离性；一旦构造函数执行完毕，x和y就是只读的了，各个线程只能通过GetX()和GetY()访问它们，从而保证了不变性。

例7：

```
public class Coordinate
{
   private double x, y;
    public Coordinate(double a,
                      double b)
   {
      x = a;
      y = b;
   }
   public void GetX() {
      return x;
```

```
    }
    public void GetY() {
       return y;
    }
}
```

对于文章最开始提到的关于thd->proc_info的Bug，我们则可以通过把S1和S2两条语句用锁保护起来解决。被锁保护起来的临界区在别的线程看来就是“原子”的，是不可以被打断的。

例8：

```
    线程1
线程 2
LOCK(&lock)
S1: if (thd->proc_info)
LOCK(&lock);
{
S3: thd->proc_info=NULL;
  S2: fputs(thd->proc_info,...)
UNLOCK(&lock);
}
UNLOCK(&lock)
```

从Java 5 开始，我们可以使用并发库来简化并行编程。例如，ConcurrentLinkedQueue中的add()、poll()等方法都是原子的、线程安全的，在单独调用这些方法时我们是不需要自己进行同步的。但是，两个原子的操作组合在一起可就不是原子操作了，例如：

例9：

```
Queue<String> queue = new ConcurrentLinked
Queue<String>();
...
if(!queue.isEmpty()) {
   queue.poll(obj);
}
```

在执行isEmpty()之后、poll()之前，这个queue有没有被其他线程修改是不确定的，所以在这种情况下，我们还是需要自己同步，用加锁的方式来保证原子性（虽然这样很损害性能）：

例10：

```
synchronized(queue) {
    if(!queue.isEmpty()) {
       queue.poll(obj);
    }
}
```

违反顺序性

简单来说，多线程程序各个线程之间交错执行的顺序的不确定性（Non Deterministic）是造成违反执行顺序Bug的主要根源（严格来讲，多线程交错执行顺序的不确定性只是违反执行顺序Bug的原因之一。另一个可能造成违反执行顺序Bug的原因是编译器/CPU对代码做出的违反多线程程序语义的乱序优化，在上一期的《浅析C++多线程内存模型》中我们有对乱序优化做过介绍）。正是因为这个原因，程序员在编写多线程程序时就不能假设程序会按照你设想的某个顺序去执行，而是应该充分考虑到各种可能的顺序组合，从而采取正确的同步措施。

举例来说，下面这个来自Mozilla的多线程Bug产生的原因就是程序员错误地假设S1一定会在S2之前执行完毕，即在S2访问mThread之前S1一定已经完成了对mThread的初始化（因为线程2是由线程1创建的）。事实上，线程2完全有可能执行的很快，而且S1这个初始化操作又不是原子的（它需要几个时钟周期才能结束），从而在线程1完成初始化（即S1）之前就已经运行到S2，最终导致S2中对mThread->State的读操作出现错误。

例11：

```
    Thread 1
Thread 2
void init(...)
void mMain(...)
{ ...
{ ...
 S1: mThread=
...
      PR_CreateThread(mMain, ...);
S2: mState = mThread->State;
  ...
...
}
}
```

上面这个例子是一个线程读、一个线程写的情况，除此之外还有违反写—写顺序以及违反一组读写顺序的情况也可能产生违反顺序性的Bug。例如下面这个程序，程序员错误的假设S2（写操作）一定会在S4（也是写操作）之前执行。但实际上这个程序完全有可能先执行S4后执行S2，从而导致线程1一直阻塞在S3处：

例12：

```
Thread 1 Thread 2
int   ReadWriteProc(...)
void DoneWaiting(...)
{
{
                    .       .       .
/*callback func of PBReadAsync*/
 S1: PBReadAsync(&p);
  S2:  io_pending  =  TRUE;
```

```
...
                    .       .       .
S4: io_pending = FALSE;
  S3:  while  (io_pending)  {...}
...
                    .       .       .
}
}
```

下面这个是违反一组读写操作顺序的例子：程序员错误地假设S2一定会在S1之前执行，但事实上可能S1在S2之前执行，从而导致程序crash。

例13：

```
Thread 1 Thread 2
void js_DestroyContext(...){
void js_DestroyContext(...){
  /* last one entering this func */   /*
non-last one entering this func */
  S1: js_UnpinPinnedAtom(&atoms);
S2: js_MarkAtom(&atoms,...);
}
}
```

调试违反执行顺序这类的多线程Bug最困难的地方就在只有某几种执行顺序才会引发Bug，这大大降低了Bug重现的几率，增加了调试的难度。最简单的调试手段自然是使用printf了，但类似printf这样的函数会干扰程序的执行顺序，又增加了该类Bug复现的难度。

违反顺序性的解决方案

下面我们介绍一下违反顺序性Bug的常见解决方案，以帮助大家避免和修正此类Bug。

（1）加锁进行同步

加锁的目的就在于保证被锁住的操作的原子性，从而这些被锁住的操作就不会被别的线程的操作打断，在一定程度上保证了所需要的执行顺序。例如在例12中，我们可以给{S1,S2}一起加上锁，这样就不会出现S4打断S1、S2的情况了（即S1->S4->S2），因为S4是由S1引发的异步调用，S4肯定会在{S1,S2}这个原子操作执行完之后才能被运行。

（2）进行条件检查

进行条件检查是另一种常见的解决方案，关键就在于通过额外的条件语句来迫使该程序会按照你所想的方式执行。例如下面这个例子就会对n的值进行检查：

例14：

```
retry:
n = block->n;
...
...
if (n!=block->n)
{
  goto retry;
}
...
```

（3）调整代码执行顺序

这个也是很可行的方案，例如上面的例12不需要给{S1,S2}加锁，而是直接调换S2与S1的顺序，这样S2就一定会在S4之前执行了！

（4）重新设计算法/数据结构

还有一些执行顺序的问题可以通过重新设计算法/数据结构来解决。这个就得具体情况具体分析了。例如MySQL的bug #7209中，一个共享变量HASH::current_record的访问有顺序上的冲突，但是实际上这个变量不需要共享，所以最后的解决办法就是线程私有化这个变量。

总结

多线程Bug确实是个非常让人头痛的问题。写多线程程序不难，难的在于写正确的多线程程序。需要注意的是，违反执行顺序和违反原子性这两种Bug虽然是相互独立的，但是两者又有着潜在的联系。例如，文中的例1其实是因为执行顺序的不确定性造成的，而本文的例12就可以通过把{S1,S2}加锁保证原子性来保证想要的执行顺序。其实归根结底，我们最终是想让多线程程序按照你的意愿正确地执行，所以在清楚什么样的情形可能让你的多线程程序不能按你所想的那样执行之后我们就能有意识避免它们了（或者更加容易的修复它们）。多线程的Debug现在仍然是学术界的研究热点，工业界和学术界都在通力合作推进多线程调试技术的发展，相信在不久的将来我们会有更多更易使用的多线程Debugger帮助我们调试多线程Bug。



陈冠诚

IBM中国研究院研究员，瑞典查尔姆斯理工大学（Chalmers）硕士，主要研究方向为并行计算、计算机体系结构及分布式系统。个人博客为并行实验室（http://parallellabs.com）。

责任编辑：高松（gaosong@csdn.net）

十年一剑——Drupal 7

文 / 方擎

历经十年发展，Drupal 7成为了无数家网站所使用的内容管理系统框架，并屡获开源大奖，本文将分享Drupal 7令人激动的新特性，并带您领略十年铸成的Drupal之剑的独特魅力。

2011年1月7日，Drupal 7发布了。这是一个让所有Drupal爱好者激动的日子，数以千计的开发者连续三年的努力终于使Drupal 7露出真容。从Dries Buytaert发布了第一个版本到今天，Drupal刚好走过了十年。十年之前，它只是一个供大学实验室里使用的简单的BBS程序；十年之后，它发展成为被成千上万的网站所使用的内容管理系统框架，并屡获开源大奖，这其中离不开全世界众多开源爱好者的共同努力。本文将与大家一起分享Drupal 7的令人激动的新特性，领略十年铸成的Drupal之剑的独特魅力。

Drupal 7简介

Drupal是一个用PHP开发的开源的内容管理系统（Content Management System），包含了当今大多数网站应具有的“基础设施”——用户及权限管理、页面及文章管理、内容分类管理、导航管理以及站点信息设置等。其优良的模块化架构和丰富的API，也被用于构建除内容管理外的其他应用类型的网站，因此被认为是一个优秀的“内容管理框架（Content Management Framework）”。重要的是，与其它CMS相比，Drupal的最大优势在于拥有数以

图 1 Drupal.org首页



万计的“开箱即用”的第三方模块，使开发者能够通过组装模块实现大部分的网站功能，有效地减少了开发成本。

Drupal在近几年可谓风光无限，使用它的个人、公司和组织越来越多。从美国白宫（whitehouse.gov）到经济学人（economist.com）；从逝去的流行天王迈克尔杰克逊（michaeljackson.com）到联合国世界野生动物基金会项目（adaptationportal.org）；从Yahoo! Research（research.yahoo.com）到美国在线公司（corp.aol.com），它们均由同样的核心提供着运转的动力——Drupal。

经过了十年的磨砺，Drupal 终于发布了它第7个版本。这个版本的研发时间最长，测试时间最长，参与开发和测试的人也是历代版本最多，因此可以说它是Drupal发展历史上的一个重要的里程碑。它保留了5代和6代的较为成熟的基础功能和模块式架构，将一些第三方模块的明星功能也吸收成为核心功能，并在用户体验方面也作了很多改进和革新，使Drupal 7达到了一个新的高度。同时，在底层机制和开发接口方面也做了大量的调整和改进，使开发者能更加灵活地定制Drupal。

新界面、新感觉

为了改变视觉效果和用户体验上的劣势，做到“简约而不简单”，Drupal 7带来了两个全新制作的主题——Bartik 7.0和Seven 7.0，分别针对普通用户和管理用户设计。在Drupal中，可以同时为网站设定两个主题模板，一个是前端内容展示时所用，另一个是后台管理页面使用。管理操作相关的页面使用较为简洁且开销小的主题，大大减少了页面加载时间，提高了管理效率。

Bartik 7.0主题基本上延续了Garland主题的布局和色调，又融入了流行的黑色酷雅风格，使Drupal的外观更为简约时尚。博客、评论、菜单等最终用户（end-user）使用的界面都做了相应的优化，改善了用户体验。Seven 7.0主题则通过重新布局表单和管理相关的页面元素，将常用管理的链接放在页面顶端的“管理工具条”中，并引入“Overlay”技术，以提高管理效率。在浏览某篇文章时，通过管理工具条上的“Add Content”打开创建内容窗口，这种将管理窗口以“层”的形式显示在当前的页面上的技术被称为“Overlay”。结束操作并关闭窗口后，即恢复到原页面，避免多次的页面跳转。

除了推出了全新的主题外，Drupal 7在主题引擎方面也做了很大的改进。比如Drupal 7修改了节点模板（node.tpl.php）中的内容变量（$content），使之由原来的HTML字符变成了一个包含节点文字内容、链接、评论等多个内容的数组，并采用了“细粒度渲染（Granular Rendering）”的方式，使开发者可以通过hide和render函数控制各个元素的显示，为开发者提供了更大的自由度。此外，它还在内核中集成了jQuery UI，不用再安装第三方模块即可使用丰富的jQuery界面效果了。

重装上阵的Fields API

在Drupal中，每一个完整的内容被称为“节点”，每个节点属于一个“内容类型”。内容类型可以理解为是面向对象设计概念中的“类”，它定义了节点的属性。在以前的版本中，如果不使用第三方模块——CCK（Content Construct Kit），那么不同的内容类型之间的区别也只是一些是否置顶、是否开启评论等设置，且基本属性永远只有两个：标题和内容。

CCK模块提供了为内容类型添加更多属性的功能，极大地扩展了Drupal的应用领域范围。 它允许管理员通过页面操作即可为内容类型创建新的字段，而不需要写任何代码。字段的定义和值均存储在数据库中，可随时更改或删除。每种类型的字段均有相应的表单控件（widget），如文本型字段则会在内容编辑表单中增加一个文本框，图像字段则增加一个文件上传框等。

在经过几年的发展后，Drupal 7将CCK集成进了内核，使其作为Drupal的原生功能。而其中最核心的就是Fields API。Fields API提供了程序级别的Fields开发接口，管理页面由Fields UI模

块来实现，它们共同完成了CCK的完美替换。

在Drupal 7中，“实体（Entities）”是网站的基本数据单元。每个单独的节点（node）、用户（user）、评论（comment）或分类词条（term）均属于一个实体。一类实体的集合称为“集（Bundle）”。内容类型（Content Type）、用户（User）、评论（Comments）均

图 5 Drupal 7新官方主题——Bartik 7.0



图 6 Drupal 7的Overlay效果



属于“集”的定义范畴。Drupal 7的Fields功能比CCK更为强大，最突出之处就在于它可以扩展除节点外的其他实体。

比如，可以将Article这个内容类型增加一个图型字段，用于上传和显示文章的配图；也可为评论增加一个整数型字段，让用户在发表评论时也可以打分。如果觉得下拉框不是很友好，那么可以自己开发一个新的类型的Field和Widget，用五颗星的形式接收用户输入并展示（可参考Fivestar模块）。值得一提的是，如果创建了文本型字段，那么这个字段值可以保存为多种语言，极大地满足了网站的本地化需求。图7所示为Article内容类型的Fields编辑页面，图8所示为通过拖拽可以调整Fields的显示顺序。图9展示了调整顺序后某Article节点的显示页面。

当然，图9只是默认的节点展示页面，要实现自定义的节点展示，可以通过修改模板文件来实现。

Fields API可以说是Drupal 7中最引人注目的功能，它使得网站中的全部实体均可以被定制，极大地丰富了网站可存储的内容。同时，它也为其它模块能够使用Fields功能提供了丰富的接口，较原来CCK模块有了更强的可扩展性和交互性。可以说，掌握Drupal 7开发的关键之一，就是掌握如何使用Fields API / UI将业务需求中的数据模型转化为Drupal网站中的实体。

更强的数据库支持

为了提供更安全、更强大的数据库支持，Drupal 7对数据库抽象层作了重大的变革——全面使用PDO（PHP Data Objects）。PDO是PHP的一个扩展（extension），提供统一的、面向对象的数据访问（data-access）抽象层。不管使用的是哪种数据库，都可使用相同的PHP函数来查询或更新数据，大大增强了代码的可移植性。在7版本之前，Drupal只内置支持MySQL和PostgreSQL，而现在新增加了内置的SQLite支持。使用SQLite可将数据存放在文件中，并支持基本的SQL语句操作，且无须数据库服务器的支持。

Drupal 7的数据库抽象层具有三个优点。第一是简化了程序的数据库操作代码。如在需要进行数据分页的情况下，使用pager_query来执行查询，并结合主题函数theme_pager，可以轻松实现分页。第二是使数据库操作代码具有可移植性。不管使用的是哪个数据库，都可以使用pager_query来进行分页查询。第三是更好地保障数据库操作的安全。像db_query和pager_query等SQL语句执行函数使用了PDO的SQL语句替代符（placehoder），将查询时所用的变量通过函数的参数传递进去，有效防止了SQL注入。下面是使用db_query查询节点信息的一段示例代码。

```
$uid = 1;
$result = db_query('SELECT n.nid, n.title,
n.created FROM {node} n WHERE n.uid =
:uid', array(':uid' => $uid));
foreach ($result as $record) {
   // $record为节点（node）实例，可使用
$record->title, $record->created等属性
}
```

图 7 Fields操作界面



图 8 调整Fields的显示顺序



图 9 增加了Subtitle和配图字段的Article节点



这段简单的代码中包含了Drupal开发中最常用的数据库操作。db_query函数负责执行SQL查询，并返回结果集。SQL语句里的“:uid”即为替换符。其值是通过参数array(‘:uid’ => $uid)传递进去的。在生成最终的SQL语句前经过了特殊字符过滤等安全处理。

此外，Drupal 7还增加了对事务（Transaction）的支持。使用db_transaction函数即可创建一个事务变量。在这个变量的生存周期内进行的操作，均属于事务的一部分。当变量周期结束后，事务就自动提交了。如果有异常，可以使用rollback方法回滚事务。这个功能对于不支持事务的数据库是一种简易的解决方案。下面是Drupal 7中node_save函数的部分代码。

```
function node_save($node) {
  $transaction = db_transaction();
  try {
  …
  }
catch (Exception $e) {
    $transaction->rollback();
    …
    throw $e;
  }
```

从代码中可以看到，只有当所有的保存操作成功后，节点数据才会真正提交到数据库中，较之前版本的节点保存操作在保障数据的可靠性上有了很大的改进。

结语

本文介绍了一些Drupal 7的全新特性，但其改变远远不止于此。在安装、安全、性能、图像处理、可用性以及测试支持等诸多方面，Drupal 7都做了很大的改进与增强。在系统架构和API方面也有重大的革新。Drupal 7的发布使Drupal超越了传统的内容管理系统的局限，向更高级的网站应用程序框架迈进。你可以不使用它，但一定要去了解它，因为它的出现，势必会改变PHP开源内容管理系统的格局，并对未来的网站开发趋势产生深远的影响。



方擎

资深软件工程师，国内第一本Drupal中文图书《Drupal高手建站技术手册》作者。目前在深圳亚鼎公司工作，从事Drupal、PHP及.NET在电子商务网站领域的开发和研究。

责任编辑：高松（gaosong@csdn.net）

改变Web开发格局的新技术node.js

文 / 赵昆

node.js是近两年兴起的新技术，用于高性能后端服务的开发，现已成为开源社区的明星。本文介绍了node.js的诸多优势及发展趋向。

node.js是这两年才兴起的新技术，使用JavaScript语法，用于高性能后端服务的开发。说node.js，必须提到JavaScript，JavaScript是一个生命力非常顽强的语言，最开始由Netscape开发，本来中规中矩，却意外借助Ajax大红大紫了起来，HTML5的到来，更是为JavaScript展现了美好的未来。我们认识的JavaScript只是一个动态脚本语言，用于编写Web前端程序，而node.js的问世，让JavaScript可以前后端通吃。node.js 在2009年才发布第一个版本，很快就成为了开源社区的明星，现在在国外已经越来越热，由于JavaScript已经在前端处在垄断性地位，如果平滑移植到后端，加上node.js极佳的并发性能，所以其极有可能会成为改变目前Web开发格局的新兴力量。

背景

node.js的创始人是Ryan Dahl，他以前经常用各种语言写一些事件驱动的服务端程序，但是通常还需要依赖于Nginx等WebServer。他后来决定自己写一个不依赖于WebServer的无阻塞、事件驱动框架系统，由于JavaScript先天的事件驱动的语言特性，因此选用JavaScript作为其运行语言。2009年1月开始构想，5月就有了第一个版本。经过近2年的发展，目前最新版本是0.4.6（截止2011年4月18日），当前的版本已经很稳定，估计3~4个月之内很有可能会发布第一个正式版本V1.0，这将会如同Nginx发布1.0一样，让人充满期待。

三大优势

单线程、无阻塞、事件驱动是node.js最大的特性，为了便于理解，让我们看一段典型的代码：

```
var fs = require('fs');
var http = require('http');

http.createServer( function(request, response) {
    fs.readFile('./response.txt', function(err, data) {
        if ( err ) throw err;

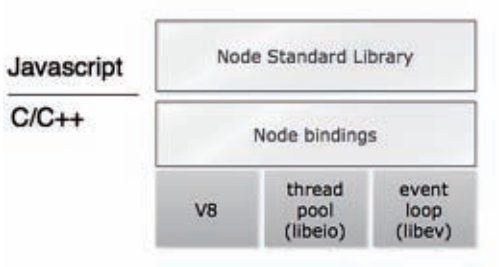
        response.writeHead(200, {'Content-Type': 'text/plain'});
        response.end(data.toString());
    });
}).listen(8080, '127.0.0.1');
```

运行这个代码就可以启动一个性能非常高的Web服务，用户输入http://127.0.0.1:8080就会显示服务器上response.txt文件的内容。

首先它是单线程的，代码运行起来就是一个独立线程，基于事件驱动来接受客户端的请求，而所有的I/O都是非阻塞的，包括通信和文件操作。这个实例里面，文件读取的过程中，只有读取到了内容，才会执行回调函数中的输出代码。这和我们传统的服务器开发模式已经有了很大的不同，不过这样的设计非常的精妙，如同Erlang中的强制函数式编程保证了分布式大型系统开发的高效，而node.js需要的异步编程，也保证了高并发网络服务程序的高效，所有通信都是事件驱动的，所有的I/O都是异步非阻塞的。

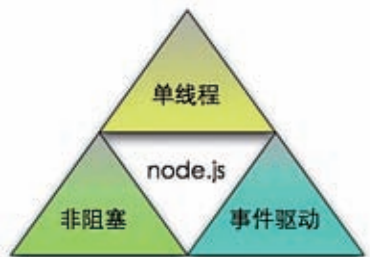
node.js的脚本执行依赖于V8，异步I/O依赖于libeio库，事件驱动依赖于libev库，node.js自身非常简单。

图1 node.js基础架构



相比于其他开发语言，“开发效率”、“高性能”和“成本”是node.js最大的的优势，JavaScript的代码可以同时运行在前端或者后端，node.js的开发非常简单，只需要几行代码就可以写出一个简单的服务器程序，同时由于它的非阻塞特性以及异步的事件驱动模型，在开发高并发网络服务时，性能非常突出。熟悉Nginx的人都知道，事件驱动系统架构的性能优势很强，但是用C语言开发Nginx模块的成本非常高，而用FastCGI挂载PHP，也会受到进程的限制，如果某些PHP执行超时，将阻塞整个服务。 由于node.js的运行不需要WebServer，自己本身是单线程的服务，所以没有多线程服务的线程通信和锁开销，并且由于事件驱动模式，维持网络连接的堆分配占用内存很小，因此node.js的系统资源占用也非常低，这样保证了node.js的运行成本，单台服务器可以用极低的系统消耗轻易处理上万个并发请求和数据处理。

图2 node.js三大优势



若干不足

当然node.js也有它不足的地方，首先由于基于异步编程模型，这与传统的串行编程有很大不同，熟悉Java、C/C++以及PHP等的开发人员刚开始可能会有些不适应，虽然这对于经常使用Ajax或者jQuery/YUI的前端开发工程师来说，已经非常熟悉。同时由于node.js历史非常短，目前大公司对于node.js支持和应用也不多，这在一定程度上影响了node.js使用者的信心。node.js的单线程特性，导致程序出现异常并导致整个服务crash，这也给系统的稳定性带来一定挑战，并且单线程的服务也使其自身无法直接有效地利用多核的CPU计算资源。虽然这些都可以有方法解决，不过这些问题仍然会给开发者造成一些困扰。

正是由于这些优势和不足，它特别适合实时、高并发的应用场景，比如：数据中间层系统、实时监控系统、Web游戏，高性能服务器，轻量级的Web应用，网络爬虫、基于内存的K/V服务等。但node.js并非为了解决所有问题而生，所以千万也不能把它想象成一个万能技术。

发展前景

同时node.js采用JavaScript语法，其解释引擎基于V8，V8也是Google Chrome的JS引擎，效率非常高。得益于V8的高性能以及node.js自身简洁优良的体系架构，node.js的发展速度非常的快，代码已经有了20多个活跃的开发者。

今年2月，淘宝几个对node.js很感兴趣的工程师成立了CNodeJS.ORG组织，目前社区还非常不活跃，并且缺乏有效的组织，很多时候还需要做一些普及的工作。我们认为node.js如果发展成熟，未来完全可能取代LAMP以及Nginx+FastCGI+PHP的黄金组合。

目前已经有越来越多的技术牛人开始关注node.js，而JavaScript的使用人群非常广泛，大家都知道JavaScript的自身语言生命力非常强，号称比小强还强，这也意味着node.js也具备更强的生命力基础，我们非常看好这个技术的未来。



赵昆

CNodeJS.ORG开源社区的发起人之一，来自淘宝数据平台与产品部，花名空无。热爱技术，多年来持续关注海量数据处理、Web开发、数据可视化相关的技术。同时也是微博控+数据控+苹果控。

责任编辑：高松（gaosong@csdn.net）



Drive Safety随身便携酒精检测小精灵

虽然身材不大，却也是有着“高科技”内涵的小家伙，平时在汽车钥匙链上当个看时间的表，和朋友聚会不小心喝了那么几口的时候，赶紧揪出来吹口气，看看是不是超标了，为自己也为家人安全着想。使用非常简单，小处有大智慧的表现，内置先进的半导体酒精传感器，能够相对准确地测试人体血液中酒精含量范围，并用LED灯分三档显示。



BodyGard PTX 3合1求生工具组

这款SWISS+TECH研发的最新产品将汽车救生工具与超亮LED灯结合为一体，制造出这款具抗泼水、耐用且精美的BODYGARD迷你三合一紧急防身工具，重量仅仅不到30g。相比较SWISS TECH的其他救护系列产品，本产品功能组合变得简单纯粹（只有3种功能），体积重量较先前的救护系列更加轻便，功能虽然少了几样，但是专业性却变得更强，正所谓业精于细！

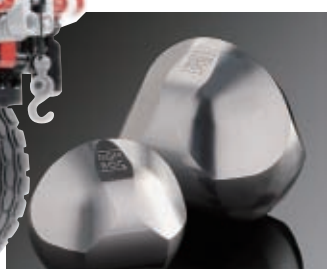
GEEK产品

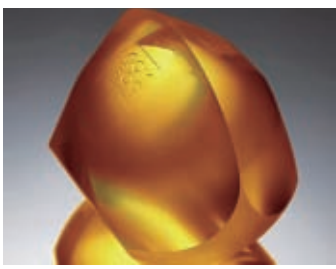
*感谢麦极网对本栏目的支持



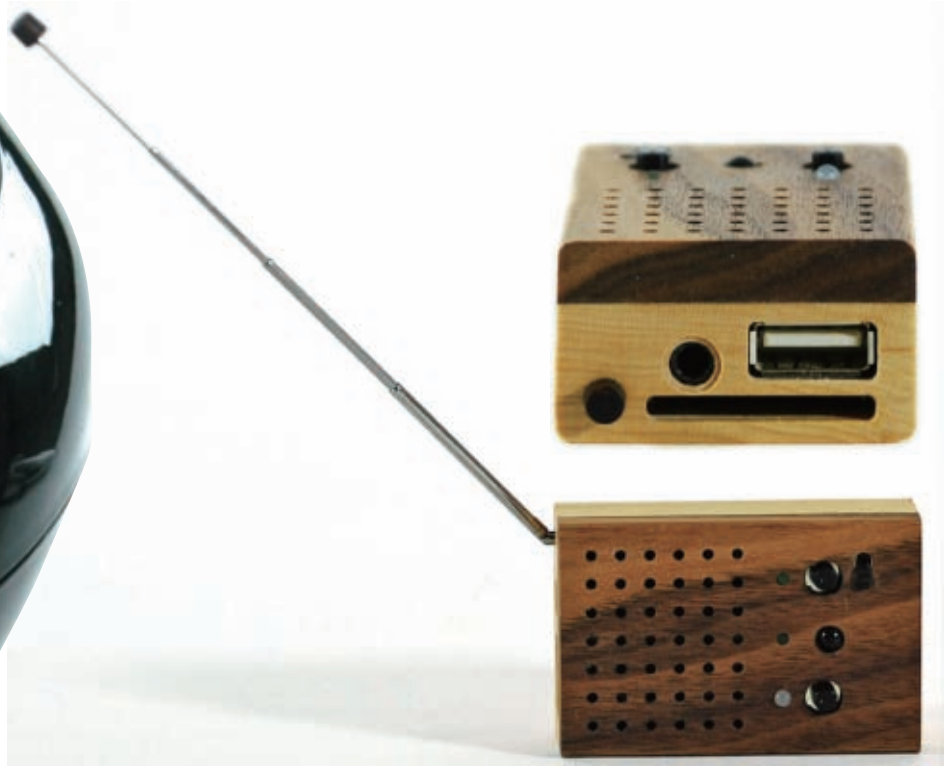
Mercedes-Benz Unimog

适逢Mercedes-Benz Unimog六十周年，乐高特意为Unimog六十大寿推出型号为Unimog U 400的积木吊臂车，具备逼真的还原程度，以1:12.5的比例从真实吊臂车模拟过来，许多细节如引擎到活塞都忠实呈现。车身上还有气泵马达，可进行各种和真实Unimog同样的动作，车尾的吊臂配件也能做出360度旋转动作，车内的每项功能也效仿真实吊臂车的正常运作而成。









Motz Mini FM Radio

韩国Early Shop出品的一款超小型收音机，全木质外壳，仅手指大小。除了能收听广播，还可以作为外置音箱用。支持USB连接和SD卡存储。充满电后，作为外置音箱可工作5小时，收音机功能可工作10小时。

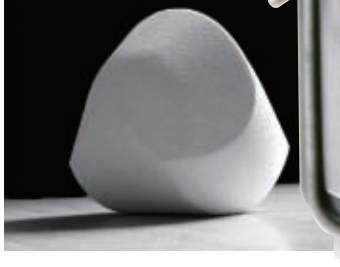
Hand-Held 2.4G手握式轨迹球无线空中鼠标

谁说鼠标一定要放在桌子上使用？这款Hand-Held Multipurpose2.4G手握式轨迹球无线空中鼠标就可以让你在任何地方，空中挥舞，来一个捕风捉影式的操作，超高灵敏度感应的轨迹球绝不亚于任何光电鼠标的准确性！从此告别鼠标手！

Gomboc

Gomboc是匈牙利布达佩斯工艺与经济大学的两位科研人员发现的，世界上首个只有一个稳定平衡点和一个非稳定平衡点的均质物体，它最大的特点是无论以何种角度将其放置在水平面上，都可以自行回到其稳定点。现有金属、玻璃等不同材质的模型。





Hirokawa第二代手提式复古音乐盒

还记得小时候的铁盒子吗？铁皮曾经一度占据了我们大部分人的少年时光，现在，动感铁盒子音箱问世啦，它不但能当随身小音箱，还可以作为手机和MP3的收纳盒哦！



新书上架

云计算：技术、应用、标准和商业模式



编著：周洪波

出版社：电子工业出版社

作者早年在美国Oak Ridge国家实验室、IBM和DoubleTwist等企业工作，期间多从事并行和分布式HPC超级计算技术相关工作，近年来，又多研究和采用了基于多租户模式的SaaS技术作为M2M/物联网运营平台的支撑，并且2007年实地考察了Salesforce的APEX PaaS技术。本书既是他多年实际工作经验的总结，也是对新形势下云计算技术的前瞻阐释。

全书共11章，探讨了和云计算相关的虚拟化、商用分布式计算、SOA；介绍了HPC系统及其发展、HPC软件与中间件；论及云计算的体系架构、典型技术与系统以及云计算的标准化和业务模式等内容。

iOS软件开发揭密：iPhone&iPad企业应用和游戏开发



作者：虞斌

出版社：电子工业出版社

作者虞斌的重要涉猎领域包括，RTOS、手机软件和信息安全，并且对Scrum敏捷开发有深入的了解。他的这些所长在本书中也得到了体现，全书体系严密，介绍的每一个实例都是项目经验的提炼，涵盖了iPhone和iPad软件开发的核心技术要点。想了解苹果产品的开发理念，掌握iPhone和iPad等软件开发设计的技术基础，本书能够提供一个全面的参考。

全书共十章，除了入门级的开发环境设置、Objective-C编程基础、iPhone SDK开发基础、iPad软件开发基础等内容的详细讲解，还结合实例（企业应用、商业软件以及游戏开发）介绍了iPhone SDK高级编程，软件发布等内容。

Linux内核设计与实现（原书第3版）



作者：Robert Love

译者：陈莉君、康华

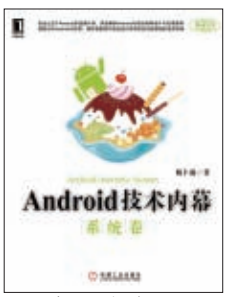
出版社：机械工业出版社

Robert Love是Linux内核核心开发人员，也是核心黑客之一。

全书主要内容包括：进程管理、进程调度、时间管理和定时器、系统调用接口、内存寻址、内存管理和页缓存、VFS、内核同步以及调试技术等。同时也涵盖了Linux 2.6内核中颇具特色的内容，包括CFS调度程序、抢占式内核、块I/O层以及I/O调度程序等。本书采用理论与实践相结合的路线，能够带领读者快速走进Linux内核世界。

作为更新的第三版，本书新增了如下内容：专门用一章篇幅描述内核数据结构；描述中断处理程序和下半部机制；扩充虚拟内存和内存分配的内容；调试Linux内核的技巧；内核同步和锁机制的深度描述；提交内核补丁以及参与Linux内核社区的建设性建议。

Android技术内幕：系统卷



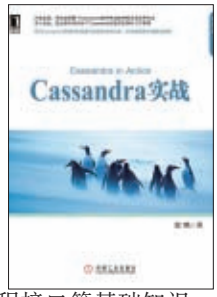
作者：杨丰盛

出版社：机械工业出版社

国内首本系统地对Android的源代码进行深入分析的著作。全书将Android系统从构架上依次分为应用层、应用框架层、系统运行库层、硬件抽象层和Linux内核层等5个层次，旨在通过对Android系统源代码的全面分析来帮助开发者加深对Android系统架构设计和实现原理的认识，从而帮助他们解决开发中遇到的更加复杂的问题。

全书按次序介绍了Android系统的架构以及各个层次之间的关系，搭建Android源代码开发环境的方法；Android的内核机制和结构；Binder的架构和工作机制；Android电源管理模块的机制与实现；Android硬件设备驱动的工作原理和实现；Android原生库的原理及实现；硬件抽象层的原理与实现，系统运行库层Dalvik虚拟机的架构、原理与实现，以及Android的核心库相关的知识。

Cassandra实战



作者：郭鹏

出版社：机械工业出版社

这是国内首本Cassandra专著。内容基于Cassandra最新版撰写，系统讲解了Cassandra的所有功能特性和使用方法，还设计了一个完整的在线交易系统实例。

全书共13章，首先简单介绍了NoSQL的优势；其次详细讲解了Cassandra的安装和配置、数据模型和排序规则、编程接口等基础知识；接着以迭代的方式演示了一个基于Cassandra的在线交易系统的完整开发过程，很好地将基础理论融入到了实践中；再接下来结合源代码分析了Cassandra的集群机制、内部数据存储结构、数据更新机制、数据读取机制、数据压缩机制、启动流程等与Cassandra的底层机制和工作原理相关的内容；最后讲解了Cassandra在分布式环境中的应用、与Hadoop的整合，以及相关的最佳实践。

Flex 4权威指南



作者：Michael Labriola、Jeff Tapper、Matthew Boles

译者：张翠芬、杨博、杜昱宏

出版社：人民邮电出版社

本书是Adobe公司的官方教程，以课程的形式逐步讲解了如何用Flex 4 开发一个完整的网上杂货商店应用程序，这个杂货商店能够动态显示数据和图像，并帮助用户从完成结账，到数据提交至服务器的整个流程。

针对Flex开发人员必须理解的概念，作者讲解了如何构建Flex应用（不应该如何构建）；MXML与ActionScript是什么关系，什么时候用哪种语言更合适；如何向Flex应用中加载数据；如何使用Flex组件，如何自定义组件；所编写代码在性能方面有什么意义；怎么才能编写出可扩展、易管理和可重用的代码等知识。

微博营销：140字的淘金游戏（第2版）



作者：Joel Comm

译者：刘吉熙、杨硕

出版社：人民邮电出版社

Joel Comm是美国企业家演讲家、畅销书作者以及网络营销大师。本书是一部详细剖析微博营销的专著。Joel在书中展示了利用微博形成一个固定而庞大的消费者群体，以取得即时商务的优势，迅速拓展品牌的影响力。

全书共12章，从社会化媒体的前景展望，到分析Twitter等微博为何如此风行，作者推荐了以正确的方式开始Twitter之旅、如何获得粉丝关注、推帖的礼仪、联系客户等经验之谈。并介绍如何利用Twitter构建品牌、调动粉丝，甚至快速赚钱的方法。作为一本网络营销书籍，甚至还介绍了如何编写Twitter程序，作者对此方面的理解深入，可见一斑。

全球排行榜

Amazon	
1	The Singularity Is Near: When Humans Transcend Biology
2	The Information: A History, a Theory, a Flood
3	In The Plex
4	Head First Java, 2nd Edition
5	The Shallows: What the Internet Is Doing to Our Brains
6	Programming in Objective-C
7	The Internet is a Playground: Irreverent Correspondences of an Evil Online Genius
8	Don't Make Me Think: A Common Sense Approach to Web Usability
9	The Clean Coder: A Code of Conduct for Professional Programmers
10	Introducing HTML5
天珑书局（中国台湾）	
1	前进Android Market！Google Android SDK实战演练
2	Google Android SDK开发范例大全
3	iPhone创意程序设计家
4	鸟哥的Linux私房菜：基础学习篇
5	iOS SDK 4 / Xcode 4应用程序开发范例集
6	HTML5在iPhone/iPad/Android的应用程序开发实作
7	Google Android 2.X应用程序开发实战
8	ASP.NET 4.0专题实务II：范例应用与4.0新功能
9	Hello！HTML5
10	深入浅出Android系统原理及开发要点
第二书店	
1	Android技术内幕：系统卷
2	云计算核心技术剖析
3	云计算（第二版）
4	Cassandra实战
5	Android应用开发揭秘
6	MongoDB权威指南
7	Lucene实战
8	Android 3D游戏开发技术详解与典型案例
9	Linux内核设计与实现
10	深入理解计算机系统

Mac OS X 背后的故事（三）

Mach之父Avie Tevanian

文 / 王越

《Mac OS X 背后的故事》系列文章将为大家介绍Mac OS X的发行版本、技术历史、相关人物等内容。本文是系列连载的第三篇。

1975年，美国罗彻斯特大学纽约分校，一组研究员正在做一个名为RIG（Rochester's Intelligent Gateway）的项目，它由Jerry Feldman主持设计。RIG的目标是给所有本地以及远端的计算设备（比如磁盘、列印机、磁带、绘图机等）提供一组统一的访问方式，其作业系统称为Aleph。为了实现所需要的功能，Aleph的内核主要构建了一个进程交互（Interprocess Communication，IPC）的机制。RIG的各进程，只要设置了目标端口，就可以彼此间发送信息。RIG项目没过几年就被判了死刑，主要是缺少很多有用的功能，比如端口没有保护机制，一次最多只能发送2KB大小的信息（受硬件限制），也没有很好的网络支持等。不过在20世纪70年代，这个系统依然代表着当时作业系统设计的先进水平，比如除了进程交互外，每个进程还有内存保护的功能，这足以让20世纪90年代末都没有做出内存保护技术的Apple公司汗颜。



Mach之父Avie Tevanian

该项目后来失败了，随后在1979年，RIG的Richard Rashid博士毕业到卡内基-梅隆大学当教授，开始做Accent项目（注：Rashid, Richard F.; George G. Robertson. Accent: A communication oriented network operating system kernel, Proceeding SOSP '81 Proceedings of the eighth ACM symposium on Operating systems principles ACM New York, NY, USA ©1981）。它是一个网络作业系统，于1981年4月开始活跃开发。受RIG的影响，Accent系统的亮点也在于可以使用IPC，而且解决了很多RIG的不足。比如每个进程有4GB的虚拟内存空间，而且甚至连内核自己都可以被存入缓存页面，内存有先进的更新前拷贝（Copy-on-Write）功能，可以实

现进程间大信息的传送等。读者可以把Accent理解为支持虚拟内存技术，并且具有网络透明IPC功能的RIG内核。

但过了几年，开发者们越来越对Accent失去兴趣。在1980年初，很多人觉得多核计算是计算机未来发展的潮流，但Accent内核在设计时并没有考虑到这些问题。而且，随着许多实验室纷纷购置性能更强劲的计算机，这就意味着Accent需要移植到新的目标架构上。此外，Unix正大行其道，不管是在作业系统理论上还是在用户程序上，都成为最为流行的作业系统模式，而Accent并不是一个Unix系统，所以无法享受Unix世界的诸多美好。为了解决这个问题，研究人员决定把所有设计推翻重来。于是就有了一个全新的系统（参见：http://en.wikipedia.org/wiki/Accent_kernel）。

在匹兹堡的一个雨天，卡内基-梅隆大学的Avie Tevanian，此系统的最主要开发者，正打着伞和同学们在去吃午饭的路上。他们一边绕着无数的泥塘，一边构思给这个新系统取什么名字好。灵感突来，Avadis Tevanian建议把这个系统叫作Muck（注：泥尘，全名作Multiprocessor Universal Communication Kernel），引得同学们哈哈大笑。后来，Richard Rashid和一位意大利同事Dario Giuse说起这玩笑，结果这位同事不经意地把Muck发为Mach，遂把Richard Rashid笑翻，伟大的Mach系统因此得名（注：Singh, Amit (2006-07-28). "A Technical History of Apple's Operating Systems". osxbook.com. p. 103. Retrieved 18 March 2011）。

Mach是一个受Accent启发而搞出的Unix兼容系统。那年，Unix已经十六岁，而且依然是作业系统理论与实践开发的主要阵地。Unix内核由于新加入的功能越来越多，变得越来越复杂。而Mach的一个主要目标就是尽量缩减Unix的各项服务，以使内核变得简单可维护。此项目从1984年开始，目标主要是包含完整的多任务支援、良好的硬件移植性，并要把大量服务移出内核作为跑在内核上的服务，以及提供与Unix的兼容性。

Mach使用纯C编写，所以在一定程度上保证了可移植性，这事实上为后面的NeXT向PowerPC移植以及2005年的向Intel移植提供了很重要的前提。而为了缩减内核该管的任务，Mach做得很绝，只提供内存和处理器管理。类似于档案系统、网络、输入输出等功能都作为单个的系统进程，独立执行于内核之上。Mach的开发过程以4.3BSD作为起点，以RIG的Accent作为参考，采纳DEC的虚拟内存设计思路，逐步开发，以新写的代码代替BSD的代码。两年后的1986年，虽然没能把系统服务完全分离于内核之外，但已颇见成效。Mach第一版大功告成，组员发表会议论文*Mach: A New Kernel Foundation for UNIX Development*，成为操作系统史上里程碑式的经典，引发操作系统业界的“微内核”学潮，如今学习作业系统设计的皆需学习此文，二十五年来被引用一千二百余次。

这篇文章主要讲了两方面内容：IPC和虚拟内存。在IPC方面，Mach把复杂的消息传送机制分为四个独立的清晰概念——任务、线程、端口、信息。任务是拥有一组系统资源的对象，允许线程在其中执行；线程是执行的基本单位，拥有一个任务的上下文，并且共享任务中的资源。

由于该论文的影响力，所以项目得到了OSF（Open Software Foundation）在内的很多投资。当然了，学术和工程永远存在差距，所以即使是最受欢迎的Mach 2.5其实仍然是一个包括大多数BSD服务层的单内核。但包括NeXTSTEP、OSF/1在内的很多操作系统都采用Mach作为其内核技术，原因是广大研究人员依然相信微内核代表着未来。虽然Mach 2.5的效率比传统的Unix系统稍低一些，但研究者们表示情绪淡定，因为Mach支持多处理器系统，可以利用多线程把任务处理得飞快，相比之下其他Unix内核并没有多处理器的完善支援，因此Mach效率稍低完全可以接受。但随着真正把Mach和BSD服务完全脱离的Mach 3微内核面世，研究人员们的情绪就再也淡定不起来了。因为服务和内核分离后，任务间的IPC数量暴涨，一个简单的Unix系统调用要涉及到十多个开端口、设权限、发

送、收取消息的操作，哪怕是使用数年后的1997年的硬件，跑一个系统调用密集的程序，Mach的效率要比一般的Unix系统慢50%，而且根本没有什么好方法来解决这个问题。

所以Mach 3出来后，虽有少数微内核信徒继续执著地改进Mach，或者开始其他微内核比如L4的研究。但学术界对Mach的兴趣大减，因而Mach 3也成为最后一版。项目解散后，Richard Rashid去了微软研究院。

再说我们的主角Avie Tevanian，他1987年博士毕业去了NeXT。这家公司刚刚由Steve Jobs成立两年，这两年Steve Jobs啥正经事都没干，只是花了十万美元雇Paul Rand设计了一个公司商标。直到Avie Tevanian加入后，这个公司才开始干实事。1987年公司确认要开发一个面向研究人员使用的计算机工作站，于是软硬件的开发工作紧锣密鼓地展开。硬件组由领导过Apple Lisa的Rich Page原班人马负责，而软件则由Avie Tevanian负责，计划开发一个有图形界面的操作系统NeXTSTEP。由于Avie Tevanian是Mach主要的开发者，自然NeXTSTEP就基于Mach了。1988年10月12日，NeXT发布预览版（0.8版），并于1989年9月18日发布1.0版（注：http://en.wikipedia.org/wiki/NeXTSTEP）。

作为NeXTSTEP系统的内核，NeXT分支的Mach经历了不少变化。NeXTSTEP 0.8主要使用Mach 2.0版，而稍后的NeXTSTEP 1.0版主要基于Mach 2.5版，包含一个自己定制的当时最新的4.3BSD服务层。但和常规的Mach不一样，它还包括一个全新的设备驱动框架，名为Driver Kit。最早在3.1版中出现（注：NeXTSTEP Release Notes, Driver Kit. http://www.cilinder.be/docs/next/NeXTStep/3.3/nd/ReleaseNotes/DriverKit.htmld/index.html），仅供x86系列的硬件使用。和Mach以及BSD代码不同，Driver Kit是使用Objective-C写的。为什么是一个面向对象的语言呢？看NeXTSTEP 3.3的DriverKit文档（注：http://www.cilinder.be/docs/next/NeXTStep/3.3/nd/OperatingSystem/Part3_DriverKit/Concepts/1_Overview/Overview.htmld/）读者大概就会发现，NeXTSTEP把所有硬件设备理解为对象，而我们知道，对象之间有继承关系，比如，磁盘（IODisk物件）属于输入输出设备（IODevice物件）的子物件，而磁盘（IODisk）本身又是逻辑磁盘（IOLogicalDisk）的父物件。硬件的初始化对应于每个物件的初始化（init方法），硬件又有读、写，所以可以用getter/setter的方法。因此，DriverKit是一个非常有特色的实现。而且由于Objective-C的效率很高，依赖很少（当时的Objective-C可以直接被翻译成C语言被C编译器编译），所以也是编写驱动的良好选择。几年后的IOKit其实就是个DriverKit的翻版。

这时，NeXTSTEP操作系统大获成功，风险投资商们纷纷购买，但硬件却始终卖不出去（注：Aaron Hillegass《Cocoa Programming for Mac OS X》前言），所以NeXT砍掉了硬件部门专做软件，更是使NeXTSTEP发展到了巅峰时期，同时支持68K、x86、PA-RISC和SPARC等硬件，但颇有意味的是它就是不支持PowerPC架构。它可以同时产生一个包含所有架构可执行码的二进制文件，来使开发的程序在所有平台上执行。这个功能也影响了后来Mac OS X的技术。Mac OS X 10.4时代有两件跨时代意义的事情，一件是Apple搞出了64位的Power Mac，开发者可以发布一个包含64位和32位程序的单一可执行文件，而无需让用户去区分；另一件是和Intel合作。Apple正式发表了Universal Binary技术，可以一个Mach-O文件同时包含Intel和PowerPC的指令。这非常贴心的设计（要知道，大多数电脑用户根本不知道Intel、PowerPC、64位、32位等技术）就是来自于Mach的技术。

NeXTSTEP 3.3后，NeXTSTEP因为NeXT和Sun的合作改名为OPENSTEP，1996年发布4.0版，到1997年2月4日，NeXT被Apple收购之前，期间内核改进除源码同步到Mach 3.0版外不明，而且出于不知道的原因，我手头的OPENSTEP正式版光盘中，居然找不到DriverKit的发布说明和编程文档，故不作详述。不过这段时间，Apple的活动值得好好一说。之前在《Linus Torvalds的短视》中，我们曾提到，1996年，Apple和OSF曾经合作，把Mach移到PowerPC Mac上，再把Linux作为单一的服务跑在Mach上，这个项目叫做MkLinux。

在1996年发布基于Mach 3.0和Linux 1.3的预览版，并更新到2002年结束其历史使命，对Mach在PowerPC的移植性上做出了重要贡献。这个PowerPC版的Mach被叫作osfmk分支（注：Singh, Amit (2006-07-28). "A Technical History of Apple's Operating Systems". osxbook.com. p. 103. Retrieved 18 March 2011），也正是现在Mac OS X中用的分支。当然了，NeXT被合并后做了大量修改。

Apple收购NeXT后，Mach被确定作为未来的操作系统核心。Avie Tevanian被选为软件开发部的总裁。合并所有项目的号角吹响后，上层的OpenStep API和老版Mac OS的部件开始合并，而Mach也经历重大变化。主要是一方面，Mach使用了osfmk分支，但依然包含4.3BSD服务；另一方面，DriverKit被IOKit取代。这是Apple走得很被动的一步。因为当时外界普遍对Objective-C不看好，逼着Apple走老版Mac OS API的老路。而Apple自己对Objective-C也很不自信，甚至想索性换用Java了事（我们以后会谈及这段不自信的历史）。所以IOKit是一个C++的驱动架构，来符合大众口味。这些改变最早在Rhapsody中出现（我们以后也会有一期Rhapsody的专题）。但由于C++是门很恐怖的语言，所以Apple又把C++给阉割了，去掉了多重继承、模板、运行时动态以及异常，让开发者使用这种对于Objective-C来说换汤不换药的Clean C++来做驱动。但公正地说，IOKit对于Driver Kit是有不少改进的，比如IOKit可以写在用户空间跑的驱动（虽然大多仍是跑在内核空间上的），因而驱动挂了而系统不会挂。另外IOKit考虑到了计算机发展的趋势，所以在电源管理、即插即用、动态加载上做得更好。

但各位也知道，C++程序得用专门的运行库才能跑，所以Mach中又加入了一个叫作libkern的库负责C++相关的功能，同时，还有一个libsa的库提供一些类似二分查找、排序等基本算法之类的功能。最后和硬件相关的还有一个叫作pexpert（Platform Expert）的库，负责收集硬件设备列表、检测机器种类（比如处理器速度等）、解析启动参数等杂活。

至此，Mac OS X的内核完全形成，形成BSD、IOKit、Mach osfmk三足鼎立的态势，并有pexpert、libkern、libsa作为基础。Apple称它的内核杰作为XNU。其代码开源，请读者移步http://www.opensource.apple.com/source/xnu/xnu-123.5/，每个部分的代码都独立存放在一个文件夹中，条理清晰，不妨一读。

由于4.3BSD已是过眼烟云，Apple后来投入大量资源扶持FreeBSD开发。2001年，Apple将FreeBSD的发起者、领军人物Jordan Hubbard收入麾下，并在Mac OS X 10.3时基本同步到FreeBSD 5的代码（注：http://osxbook.com/book/bonus/ancient/whatismacosx/arch_xnu.html）。另外，Apple的开发也同时反馈到FreeBSD小组，比如FreeBSD用的Audit（注：man audit或参见http://manpages.unixforum.co.uk/man-pages/unix/freebsd-6.2/4/audit-man-page.html）以及libdispatch（注：http://wiki.freebsd.org/GCD）等就是Apple贡献的代码。

Mac OS X早期版本不太稳定，所以会内核崩溃。10.0版本会直接像Linux或者BSD那样打出回溯信息，很不美观，所以Apple在10.2版本开始设计了一个多国语言的图片告诉用户你的内核崩溃了，以让内核崩得看起来更优雅一点。由于包含四国语言，被国内用户戏称为“四国”（注：优雅的图片见http://support.apple.com/kb/ht1392），这是XNU的Mach osfmk部分的功能。但从10.3~10.4版本开始，系统越发稳定，正常使用已很少见到内核崩溃。而且，内核提供的服务也越来越多，使得Mac OS X成为一个完善的系统。

21世纪XNU架构方面的最重大改动是支持了PPC64（10.4版本时代）、x86架构（其实本来也一直支持的，以后讲Apple的Intel迁移时详谈）、x86_64（10.6版本时代支持但不默认，10.7版本时代作为默认）和ARM架构（iPhone和iPad使用XNU内核）等多个新架构。而其中ARM架构的支持别具意义。但2006年5月31日，功成名就的Avie Tevanian离开Apple另谋发展，此时，离Apple的iPhone奇迹发生，只有不到一年时间。



王越

清华大学建筑学院毕业，现居美国。中国著名TeX开发者，非著名OpenFOAM开发者。

责任编辑：常政（changzheng@csdn.net）

奇才CTO
——Nathan Myhrvold

文 / 苏椰

这期我们要介绍的人物，实在是太有趣了。

故事要从1986年说起。当时，IBM公司推出了一套名为TopView的多任务环境，并准备把这个环境作为PC的用户界面。比尔•盖茨自然不肯容忍自己在这方面落后，也想要搞一套类似的多任务环境。就在这个时候，盖茨听说，在加州奥克兰，有一家Dynamical System Research公司开发了一套效果几乎一模一样的多任务环境，而且消耗的资源更少，运行速度更快。比尔•盖茨当时的财力已颇为雄厚（微软正是在1986年上市的），他二话不说，就把这个名不见经传的小公司给收购了。虽然后来IBM放弃了TopView，但这次收购，却给微软带来了一位天才，他就是本文的主人公——Nathan Myhrvold。

Myhrvold十四岁上大学，二十三岁时在普林斯顿拿到理论物理博士学位，随后前往英国剑桥，在史蒂芬•霍金的门下做博士后研究。在那个时期，科研工作越来越依赖计算机，物理学家经常需要编写程序。Myhrvold也发现，他对软件的迷恋程度，反而超过了物理本身。于是他放弃了物理，跑到加州，成立了前面所说的那家多任务操作环境开发公司。在进入微软公司后，他的才华和对技术的敏感性，很快就得到了比尔•盖茨的赏识。后来，Myhrvold出任微软的首席技术官，并一直为微软工作了十三年。在这一段时间里，他最广为人知的事情，就是组建了微软研究院，这些研究院成为微软的创新之源，让这个庞大无比的帝国有了年轻的心。此外，他在几个Windows操作系统等重要项目中也发挥了重要作用。

1999年，Myhrvold离开了微软，要去体验一种不同的人生了。他创办了Intellectual Ventures（简称IV）公司，专门从事知识产权尤其是专利的开发和交易。目前，该公司总共持有超过3000项专利发明，这些专利的发明者，不仅有Myhrvold本人，还有他的老上司比尔•盖茨。通过资助大学和研究机构创造专利，然后向大公司授权并收取费用，IV的年收入高达数十亿美元。

除了公司业务之外，Myhrvold还有很多不寻常的业余爱好。他曾协助比尔•盖茨写作《未来之路》，还为非营利的外星人寻找计划（SETI@home）赞助了一百万美元。而在一个名为“eGullet”的烹饪论坛上，有一个名为“Nathanm”的用户，在过去四年内，与网友们分享了大量关于“SOUSVIDE”的资料。这是一种全新的烹饪方法，主要是通过真空袋，在精确控制温度的水中进行烹饪。人们对这种方法非常陌生，因为没有资料。Nanthanm为大家带来了需要的资料，这位神秘网友后来被媒体曝光了身份，人们惊讶地发现，他竟然是微软的前首席技术官。而Myhrvold表示，他目前正在编写一本烹饪教程，指导人们更好地掌握SOUSVIDE技术，以及其他一些新型烹饪技术。

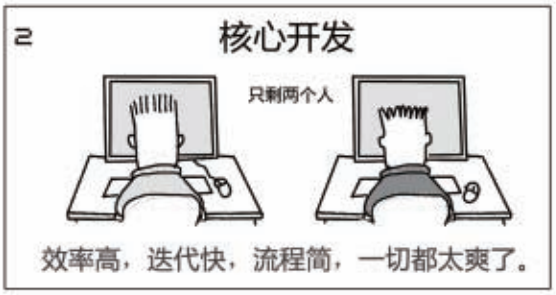


2010年，在加州举行的TED年会上，Myhrvold又带来了一个新玩意，这次他的对手不是IT公司，而是蚊子。Myhrvold表示，蚊子是世界上最烦人的东西之一，而且每年会造成2.5亿例疟疾，导致一百万人死亡，每四十三秒就有一名儿童死于疟疾。Bill & Melinda Gates基金会对抗击疟疾提出了愿景，作为其中的一项努力，Myhrvold的团队开发了一套激光灭蚊系统。在现场演示时，Myhrvold把大量的蚊子放在一个玻璃桶里，然后他的系统会自动用激光对蚊子进行追踪和击毙。这个过程的速度非常快，据Myhrvold表示，该系统每秒钟可以击毙五十到一百只蚊子。它不仅速度快，而且还非常精准，甚至能自动地区分雌蚊子和雄蚊子，因为只有雌蚊子才会盯咬人类。























Robbin:
Unix程序员。AppleFans。Geek。急需女朋友。命令行狂热者。



Hank:
Windows程序员.NET高手。已婚。保守派。



Ada:
架构师。第三性别。喜欢挖苦男人。热爱无所不能的emacs。

作者介绍：西乔
项目经理。06年起携创业团队从事Web技术外包开发及产品咨询顾问。



如果你有什么好玩的关于程序员的故事、对话、代码，愿意通过漫画的形式分享，请给西乔发邮件：arthur369@gmail.com。